

Table 11.4-10(1/2) PROBLEMS AT THE RIVER MOUTH

Serial Name	Flood Problem (Y/N)	Commercial Navigation (Y/N)	No. of Commercial Boat	No. of Fishing Boat		Fishermen Complaint Population	
				*1	*2	*1	*2
1 Perlis	N	Y	20	432 N	1	2,333 Y	
2 Baru	N	N	0	104 N	2	561 Y	
3 Sanglang	N	N	0	238 N	0	762 Y	
4 Jerlun	N	N	0	63 N	0	202 Y	
5 Kedah	N	Y	77	536 N	1	1,716 Y	
6 Yan	N	N	0	154 N	2	493 Y	
7 Melaka	N	N	0	3 N	0	10 Y	
8 Cenang	N	N	0	44 N	2	141 Y	
9 Muda	N	N	0	201 N	2	504 Y	
10 Perai	N	N	16	26 Y	0	50 N	
11 Kerian	Y	N	0	245 N	2	693 N	
12 Pinang	N	N	0	182 N	2	700 N	
13 Bayan Lepas	N	N	0	122 N	1	50 N	
14 Tg. Piandang	N	N	0	486 N	2	1,042 N	
15 Gula	N	N	0	216 N	1	308 N	
16 Sangga	N	N	0	39 N	1	76 N	
17 Larut	N	N	0	752 N	1	140 N	
18 Terong	N	N	0	5 N	0	166 N	
19 Beruas	N	N	0	655 Y	2	1,595 N	
20 Batu	N	N	0	16 N	2	21 N	
21 Dinding	N	N	0	40 N	0	83 N	
22 Lekir	N	N	0	26 N	0	55 Y	
23 Selangor	N	N	0	189 N	2	397 Y	
24 Kapar Besar	N	N	0	77 N	0	67 Y	
25 Langat	N	N	0	34 N	2	158 Y	
26 Sepang Kecil	N	N	0	23 N	2	26 Y	
27 Sepang	N	N	0	105 N	0	95 Y	
28 Lukut	N	N	0	35 N	0	79 N	
29 Raya	N	N	0	5 Y	0	10 Y	
30 Linggi	N	N	0	63 N	0	20 Y	
31 Baru	N	N	0	92 N	2	170 N	
32 Melaka	N	N	0	32 N	2	311 N	
33 Duyong	N	N	0	111 N	1	95 N	
34 Umbai	N	N	0	38 N	1	62 N	
35 Merlimau	N	N	0	35 N	1	69 N	
36 Muar	N	N	0	167 N	0	251 Y	
37 Parit Jawa	N	N	0	117 N	0	176 Y	
38 Sarang Buaya	N	N	0	35 N	0	53 Y	
39 Batu Pahat	N	N	0	64 N	0	96 Y	
40 Senggarang	N	N	0	35 N	1	53 Y	
41 Rengit	N	N	0	57 N	1	86 Y	
42 Benut	N	N	0	61 N	0	92 Y	
43 Pontian Keci	N	N	0	247 N	0	370 Y	
44 Sedili Besar	N	N	0	311 N	0	467 Y	
45 Mersing	N	Y	154	290 N	1	435 Y	
46 Endau	N	N	0	218 N	0	327 Y	
47 Pontian	N	N	0	17 N	0	28 N	
48 Rompin	N	N	0	107 N	0	405 N	
49 Merchong	N	N	0	4 N	0	11 N	
50 Nenasi	N	N	0	75 N	0	228 N	

\*1 Estimated Value? (Y/N)

\*2 0:No Compliant 1:Some complaint 2:Strong Complaint

Table 11.4-10(2/2) PROBLEMS AT THE RIVER MOUTH

Serial Name	Flood Problem (Y/N)	Commercial Navigation (Y/N)	No. of Commercial Boat	No. of Fishing Boat		Fishermen	
				*1	*2	Complaint	Population
51 Pahang	N	N	0	164	N	0	666 N
52 Terus	N	N	0	34	Y	0	86 N
53 Kuantan	N	N	45	163	N	0	570 Y
54 Beserah	N	N	0	6	N	0	21 Y
55 Kemaman	Y	N	8	97	N	0	1,338 N
56 Kemasik	N	N	0	42	N	1	175 N
57 Kerteh	N	N	23	53	N	1	140 N
58 Paka	N	N	0	83	N	0	267 N
59 Dungun	N	N	0	66	N	0	848 N
60 Mercang	Y	N	0	23	N	1	50 N
61 Marang	N	N	0	188	N	2	715 N
62 Terengganu	Y	Y	161	107	N	1	417 Y
63 Merang	N	N	0	34	N	0	66 N
64 Keluang	N	N	0	10	Y	0	39 Y
65 Gali	N	N	0	8	Y	0	15 N
66 Pak Amat	Y	N	0	28	Y	2	30 N
67 Kelantan	Y	N	0	208	N	0	666 Y
68 Rulah	N	N	0	15	Y	0	35 N
69 Sematan	N	N	0	4	N	0	97 N
70 Kayan	N	N	0	45	Y	0	104 Y
71 Sempadi	N	N	0	7	N	0	49 N
72 Rambungan	N	N	0	0	N	0	27 N
73 Sibulaut	N	N	0	0	N	0	47 N
74 Salak	N	N	0	8	N	0	54 N
75 Santubong	N	N	0	0	N	0	50 N
76 Buntal	N	N	0	5	N	0	122 N
77 Bako	N	N	0	92	Y	0	93 N
78 Sadong	N	N	0	867	Y	0	751 N
79 Kabong	N	N	0	207	Y	0	239 N
80 Oya	N	N	0	104	Y	0	292 N
81 Mukah	N	N	0	199	Y	0	556 N
82 Balingian	N	N	0	33	Y	0	92 N
83 Serupai	N	N	0	3	Y	0	7 Y
84 Tatau	N	N	0	43	Y	0	142 N
85 Suai	N	N	0	4	Y	0	12 N
86 Niah	N	N	0	4	Y	0	12 N
87 Sibuti	N	N	0	31	Y	0	99 N
88 Lawas	N	N	0	161	Y	0	167 N
89 Padas	Y	N	0	400	N	0	509 N
90 Papar	Y	N	0	123	N	1	34 N
91 Inanam	N	N	0	21	N	0	50 N
92 Tuaran	N	N	0	120	N	0	120 N
93 Bandau	N	N	0	54	N	0	54 N
94 Bongan	N	N	0	47	N	0	42 N
95 Sugut	N	N	0	211	N	0	211 N
96 Segama	N	N	0	26	N	0	28 N
97 Kalumpang	N	N	0	10	N	0	105 N
98 Tawau	N	N	0	60	N	0	400 N
99 Umas-Umas	N	N	0	15	N	0	60 N
100 Kalabakan	N	N	0	5	N	0	98 N

\*1 Estimated Value? (Y/N)

\*2 0:No Compliant 1:Some complaint 2:Strong Complaint

Table 11.4-11(1/2) CATEGORIZATION AND IMPLEMENTATION COST & BENEFIT

Serial	Name	Category	Design Boat Size GRT	Net Present Value of		B/C
				Cost RM '000	Benefit RM '000	
1	Perlis	1	150	49,395	53,581	1.0847
2	Baru	1	40	8,750	1,312	0.1499
3	Sanglang	1	40	2,394	291	0.1216
4	Jerlun	1	40	1,796	53	0.0295
5	Kedah	1	150	44,222	47,261	1.0687
6	Yan	1	40	13,067	530	0.0406
7	Melaka	3			0	*.****
8	Cenang	1	40	13,117	159	0.0121
9	Muda	1	40	6,963	106	0.0152
10	Perai	3			0	*.****
11	Kerian	1	40	2,660	132	0.0496
12	Pinang	1	40	11,391	1,086	0.0953
13	Bayan Lepas	2	40	11,243	1,484	0.1320
14	Tg. Piandang	1	40	10,738	3,193	0.2974
15	Gula	1	70	21,728	331	0.0152
16	Sangga	2	40	5,241	0	0.0000
17	Larut	2	40	9,667	199	0.0206
18	Terong	2	40	111	0	0.0000
19	Beruas	1	100	18,660	15,886	0.8513
20	Batu	2	40	17,187	79	0.0046
21	Dinding	3			0	*.****
22	Lekir	2	40	13,387	146	0.0109
23	Selangor	1	40	6,165	79	0.0128
24	Kapar Besar	2	40	8,657	437	0.0505
25	Langat	2	40	77	0	0.0000
26	Sepang Kecil	2	40	38	0	0.0000
27	Sepang	2	40	38	0	0.0000
28	Lukut	2	40	1,442	185	0.1283
29	Raya	3			0	*.****
30	Linggi	3			0	*.****
31	Baru	1	40	1,439	503	0.3495
32	Melaka	1	40	739	13	0.0176
33	Duyong	2	40	2,028	53	0.0261
34	Umbai	2	40	2,028	79	0.0390
35	Merlimau	2	40	1,407	93	0.0661
36	Muar	2	40	674	0	0.0000
37	Parit Jawa	2	40	3,064	609	0.1988
38	Sarang Buaya	2	40	3,210	0	0.0000
39	Batu Pahat	2	40	1,098	53	0.0483
40	Senggarang	2	40	2,710	53	0.0196
41	Rengit	2	40	4,304	901	0.2093
42	Benut	2	40	9,953	358	0.0360
43	Pontian Keci	1	40	4,981	2,173	0.4363
44	Sedili Besar	1	150	17,177	13	0.0008
45	Mersing	1	150	38,424	8,095	0.2107
46	Endau	1	200	21,623	18,682	0.8640
47	Pontian	3			0	*.****
48	Rompin	1	70	18,053	331	0.0183
49	Merchong	3			265	*.****
50	Nenasi	1	70	9,679	255	0.0263

\*1 1:Critical 2:Significant 3:Acceptable

Table 11.4-11(2/2) CATEGORIZATION AND IMPLEMENTATION COST & BENEFIT

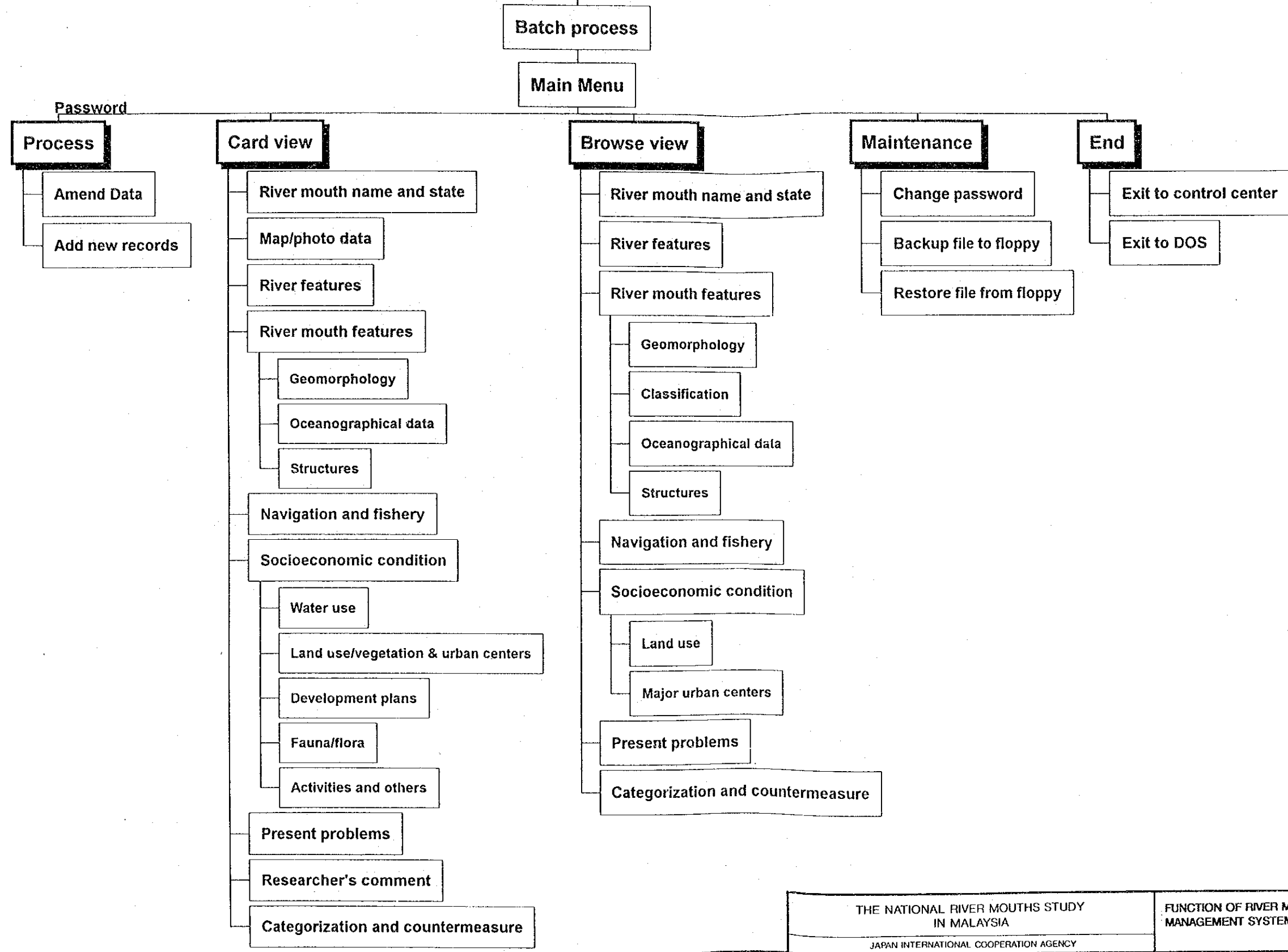
Serial	Name	Category	Design Boat Size	Net Present Value of Cost	Benefit	B/C
		*1	GRT	RM '000	RM '000	
51	Pahang	1	70	13,011	132	0.0101
52	Terus	2	40	16,057	27,479	1.7113
53	Kuantan	1	200	23,503	252	0.0107
54	Beserah	3			0	*.*****
55	Kemaman	1	100	1,913	636	0.3325
56	Kemasik	2	40	7,695	517	0.0672
57	Kerteh	2	40	8,806	40	0.0045
58	Paka	1	40	2,482	305	0.1229
59	Dungun	1	100	10,899	291	0.0267
60	Mercang	2	40	4,666	670	0.1436
61	Marang	1	40	17,335	3,670	0.2117
62	Terengganu	1	150	46,667	2,924	0.0627
63	Merang	2	40	7,553	371	0.0491
64	Keluang	3			0	*.*****
65	Gali	3			0	*.*****
66	Pak Amat	2	40	7,529	411	0.0546
67	Kelantan	1	100	9,666	2,024	0.2094
68	Rulah	3			0	*.*****
69	Sematan	2	40	5,702	13	0.0023
70	Kayan	2		3,211	0	0.0000
71	Sempadi	3			0	*.*****
72	Rambungan	3			0	*.*****
73	Sibu Laut	3			0	*.*****
74	Salak	3			0	*.*****
75	Santubong	3			0	*.*****
76	Buntal	2	40	10,544	729	0.0691
77	Bako	2	40	6,545	172	0.0263
78	Sadong	1	40	6,755	26	0.0038
79	Kabong	3			0	*.*****
80	Oya	1	40	5,134	768	0.1496
81	Mukah	1	70	31,512	729	0.0231
82	Balingian	2	40	37,044	225	0.0061
83	Serupai	3			0	*.*****
84	Tatau	2	40	16,463	1,166	0.0708
85	Suai	3			0	*.*****
86	Niah	3			0	*.*****
87	Sibuti	2	40	7,291	0	0.0000
88	Lawas	2	40	3,994	252	0.0631
89	Padas	1	40	1,513	40	0.0264
90	Papar	2	40	3,544	742	0.2094
91	Inanam	3			0	*.*****
92	Tuaran	2	40	5,354	305	0.0570
93	Bandau	3			0	*.*****
94	Bongan	3			0	*.*****
95	Sugut	2	40	4,937	0	0.0000
96	Segama	3			0	*.*****
97	Kalumpang	3			0	*.*****
98	Tawau	1	40	3,509	424	0.1208
99	Umas-Umas	2	40	14	0	0.0000
100	Kalabakan	2	40	118	0	0.0000

\*1 1:Critical 2:Significant 3:Acceptible



*FIGURES*

**River Mouth Information Management System (RMOUTH)**







Welcome to  
RIVER MOUTH INFORMATION MANEGEMENT SYSTEM  
(RMOUTH)

DID-JICA

dBASE IV V1.5

Press  to continue.

Initialize Printer? (Y/N) If Yes, on-line printer. N

THE NATIONAL RIVER MOUTHS STUDY  
IN MALAYSIA

JAPAN INTERNATIONAL COOPERATION AGENCY

WELCOME SCREEN OF RMOUTH

PRINTER INITIALIZATION PROMPT

Fig. 11.4-1

Fig. 11.4-2

Process	Card view	Browse view	Maintenance	End
---------	-----------	-------------	-------------	-----

Position with: → - <Enter> to select choice - <F1> Help

Process	Card view	Browse view	Maintenance	End
---------	-----------	-------------	-------------	-----

Enter Password:

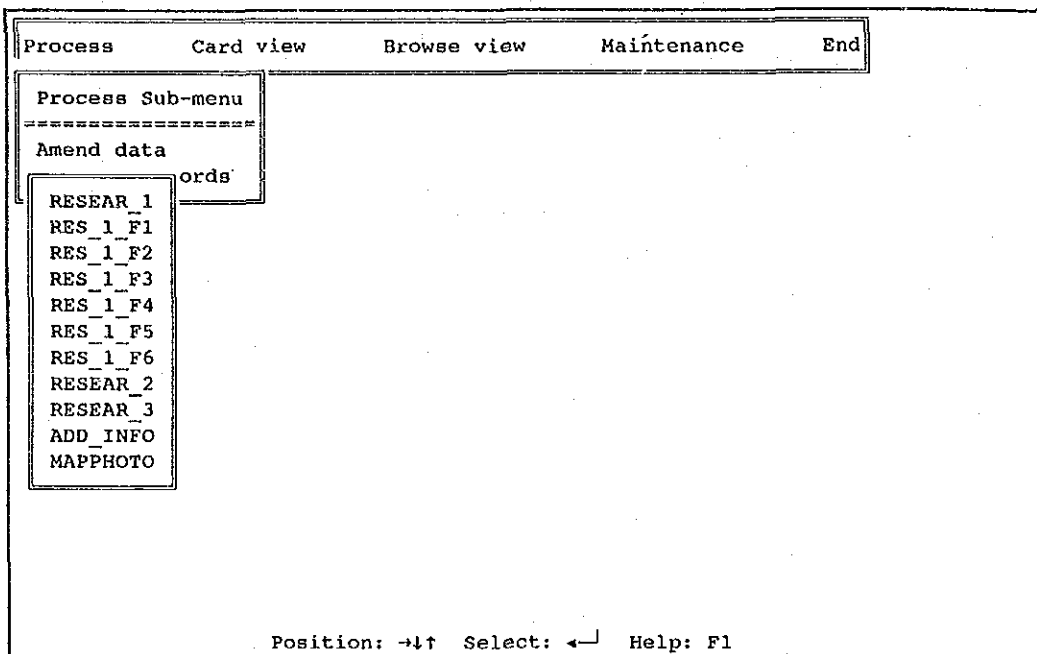
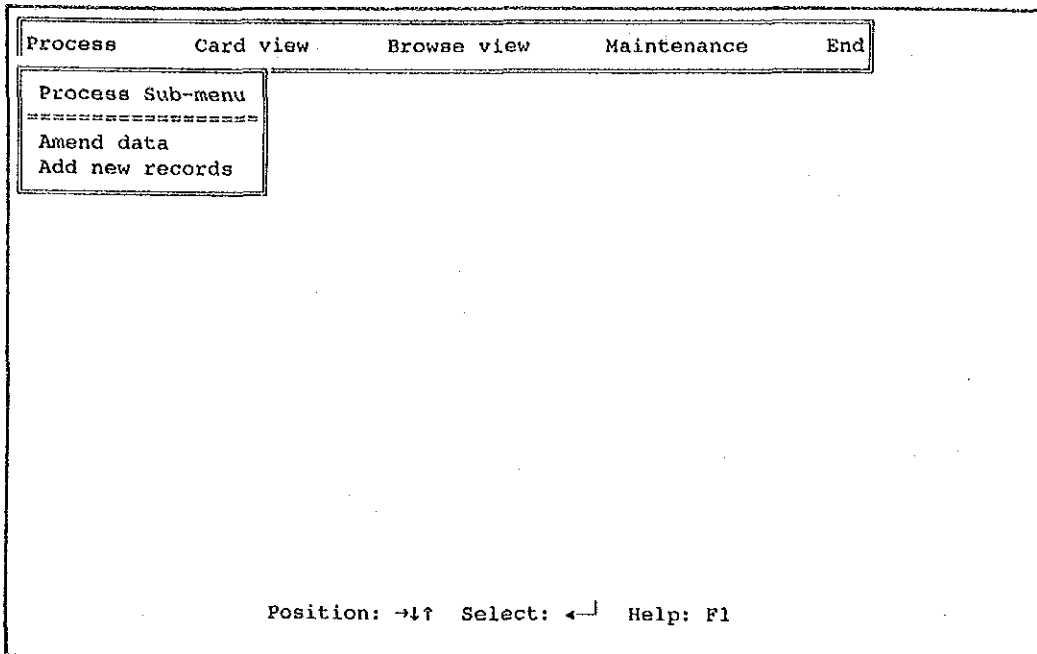
  
  
  

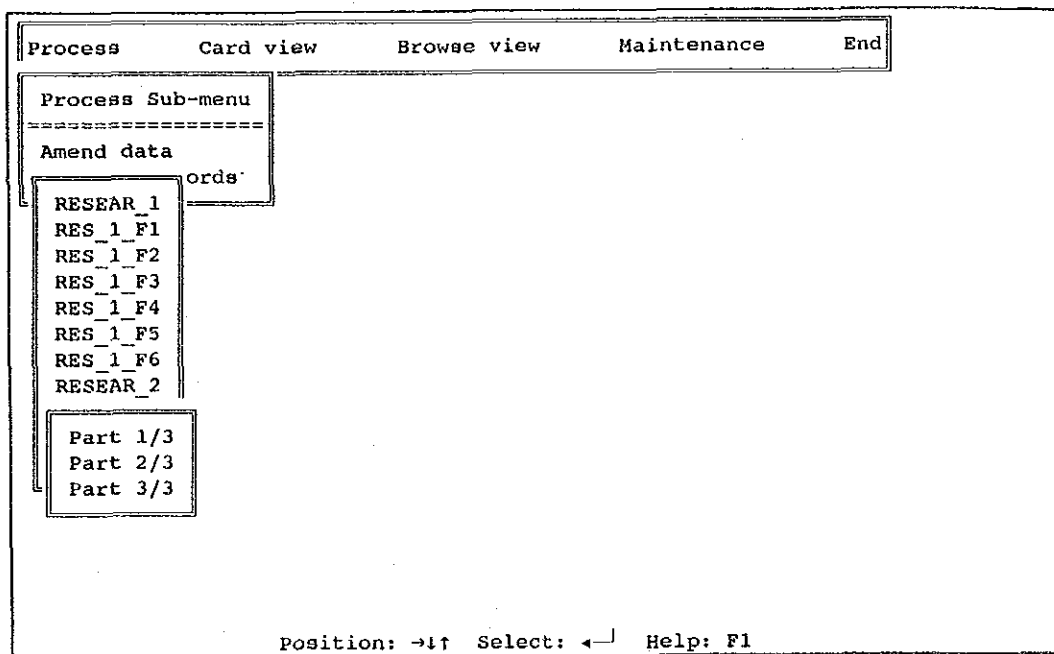
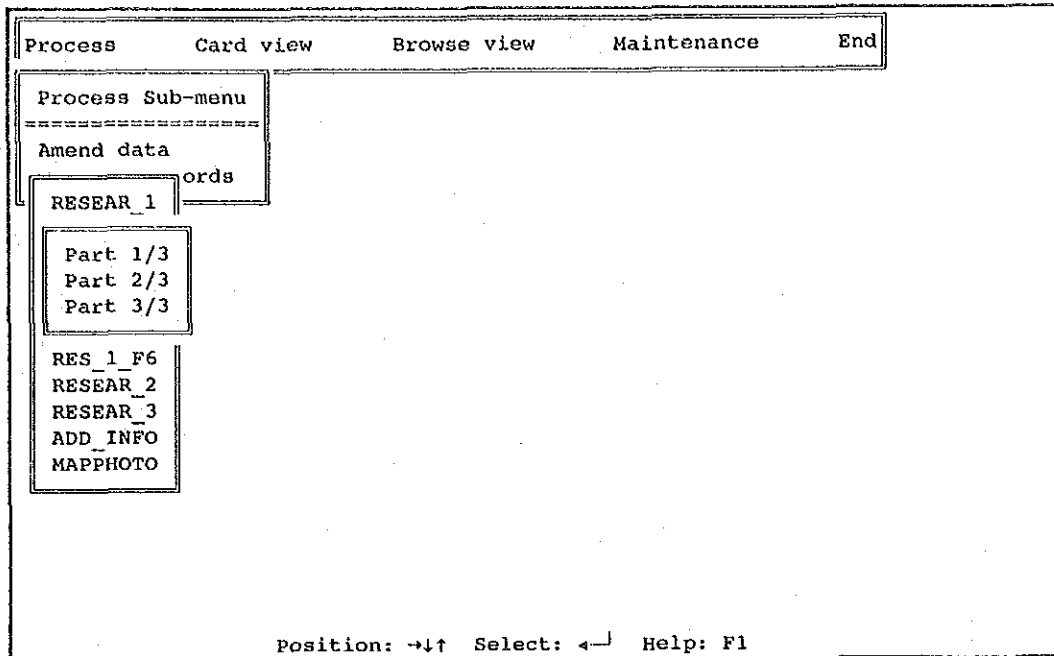
Position with: → - <Enter> to select choice - <F1> Help

Process	Card view	Browse view	Maintenance	End
---------	-----------	-------------	-------------	-----

Access denied!  
 (Password is case(uppercase/lowercase) sensitive.)





Process	Card view	Browse view	Maintenance	End
---------	-----------	-------------	-------------	-----

Process Sub-menu
=====
Amend data
ords

RESEAR_1
RES_1_F1
RES_1_F2
RES_1_F3
RES_1_F4
RES_1_F5
RES_1_F6
RESEAR_2
RESEAR_3

Part 1/4
Part 2/4
Part 3/4
Part 4/4

Position: →↑ Select: ←] Help: F1

THE NATIONAL RIVER MOUTHS STUDY  
IN MALAYSIA

JAPAN INTERNATIONAL COOPERATION AGENCY

SUB-MENU OF ITEM [RESEAR\_3] UNDER  
PROCEDURE [Amend data]

Fig. 11.4-10

Records Organize Go To Exit  
 ADDITIONAL INFORMATION (Page 1 of 7) 1: Perlis

Subject River Mouth

Serial No.: 1                      Name of River Mouth: Perlis

Edit    ||F:\dbdata\ADD\_INFO    ||Rec 1/100    ||View ||    ||    Caps

Records Organize Go To Exit  
 ADDITIONAL INFORMATION (Page 2 of 7) 1: Perlis

River Conditions

Catchment Area: 600 km2

Soil Loss (Average annual surface soil loss): 351 ton/km2/year

River Mouth Physical Conditions

Width : 513.0 m

Depth : 1.8 m

Minimum Depth: 0.6 m

Edit    ||F:\dbdata\ADD\_INFO    ||Rec 1/100    ||View ||    ||    Caps

Records Organize Go To Exit  
 ADDITIONAL INFORMATION (Page 3 of 7) 1: Perlis

Tidal Condition

Tide Port: Pulau Langkawi

Tidal Prism: Tidal Stretch: 15.0 m Average Width: 60 m

Tidal Height: 3.5 m Tidal Prism: 1,040,000 m3

Edit ||F:\dbdata\ADD\_INFO ||Rec 1/100 ||View || ||Caps

Records Organize Go To Exit  
 ADDITIONAL INFORMATION (Page 4 of 7) 1: Perlis

Existing Structures

River Mouth Improvement: / ———

Other Purposes : DM/ / ———

BW: Breakwater JT: Jetty  
 TL: Training Levee  
 RV: Revetment GR: Groyne  
 TG: Tidal Gate DM: Dam

Record of Dredging

'86, '87, '90, '91, (Schd'92) by MD

Edit ||F:\dbdata\ADD\_INFO ||Rec 1/100 ||View || ||Caps



Records Organize Go To Exit  
 ADDITIONAL INFORMATION (Page 5 of 7) 1: Perlis

Number of Commercial Boat Number: 20	Number and Size of Fishing Boat Sampan Non-powered : 0 Outboard Engine: 62  Inboard Engine 10.9 & below : 89 11 - 20.9 : 96 21 - 30.9 : 58 31 - 40.9 : 51 41 & above : 76  Total : 432 (Estimated Value Y/N): N  Fishing Boat Size : L (L:Large, M:Medium, S:Small)
Number of Fishermen Number: 2,333 (Estimated Value Y/N): Y	

Edit ||F:\dbdata\ADD\_INFO ||Rec 1/100 ||View || ||Caps

Records Organize Go To Exit  
 ADDITIONAL INFORMATION (Page 6 of 7) 1: Perlis

Land Use and Vegetation

Land Use and Vegetation Condition: UR/VI/AG

UR: Urban Area	VI: Village	AG: Agriculture	FO: Forest
SW: Swampy Area	UN: Unused Lands		

Edit ||F:\dbdata\ADD\_INFO ||Rec 1/100 ||View || ||Caps

Records	Organize	Go To	Exit		1: Perlis
ADDITIONAL INFORMATION (Page 7 of 7)					
Categorization					
Category (1:Critical, 2:Significant, 3:Acceptible): 1					
Master Plan for Improvement					
Design Boat Size: 150 GRT					
Net Present Value of Construction Cost: RM49,395,000					
Net Present Value of Benefit : RM53,581,000					
Edit	F:\dbdata\ADD_INFO	Rec 1/100	View		Caps

Records Organize Go To Exit

ADDITIONAL INFORMATION (Page 4 of 7) 1: Perlis

Existing Structures

River Mouth Improvement: /

Other Purposes : DM/ /

BW: Breakwater JT: Jetty  
 TL: Training Levee  
 RV: Revetment GR: Groyne  
 TG: Tidal Gate DM: Dam

Record of Dredging

'86,'87,'90,'91,(Schd'92) by MD

Edit ||F:\dbdata\ADD\_INFO ||Rec 1/100 ||View || ||Caps

Layout Words Go To Print Exit

ADDITIONAL INFORMATION (Page 4 of 7) 1: Perlis

Existing Structures

River Mouth Improvement: /

Other Purposes : DM/ /

BW: Breakwater JT: Jetty  
 TL: Training Levee  
 RV: Revetment GR: Groyne  
 TG: Tidal Gate DM: Dam

Record of Dredging

[.....v1.....v2.....v3.....v4.....v5.....v6.....  
 '86,'87,'90,'91,(Schd'92) by MD

Edit ||F:\dbdata\ADD\_INFO ||Line:1 Col:1 ||View || ||CapsIns

```

Layout  Words  Go To  Print  Exit
(.....v1.....v2.....v3.....v4.....v5.....v6.....).....7.....
'86,'87,'90,'91,(Schd'92) by MD

Edit  ||F:\dbdata\ADD_INFO  ||Line:1 Col:1  ||View ||  || CapsIns

```

NAME	C_AREA	SOILLOSS	RM_WIDTH	RM_DEPTH	RM_D_MIN	TIDE_PORT	T
Perlis	600	351	513.0	1.8	0.6	Pulau Langkawi	
Baru	80		100.0	0.3	0.2		
Sanglang	80		120.0	1.0	0.8		
Jerlun	40		130.0	1.4	1.1		
Kedah	4,040	415	***.*	2.3	1.8	Pulau Langkawi	
Yan	10		13.0	0.4	0.1		
Melaka	40	531	70.0	0.0	0.0		
Cenang	10		23.0	0.0	0.0		
Muda	4,300	448	200.0	3.2	1.0		
Perai	450	911	210.0	2.9	2.3		
Kerian	1,420	1,006	780.0	2.2	1.8		
Pinang	20	1,267	52.0	0.0	0.0		
Bayan Lepas	7		30.0	0.0	0.3		
Tg. Piandang	9		300.0	0.2	0.2	Pulau Pinang	
Gula	30		379.0	1.4	1.1		
Sangga	170		915.0	2.0	1.6		
Larut	170		120.0	1.5	1.2		
Terong	60		265.0	3.6	2.9		

TIDE_PORT	TIDE_L	TIDE_B	TIDE_H	TIDE_P	FACILITYRM	FACILITYOT	DRE
Pulau Langkawi	15.0	60	3.5	1,040,000	/	DM/ /	MEM
	1.0	20	3.5	23,000	/	TG/ /	mem
	1.0	20	3.5	23,000	/	TG/ /	mem
	6.3	30	3.5	218,000	/	TG/ /	mem
Pulau Langkawi	12.0	200	3.5	2,772,000	/	TG/DM/	MEM
	1.3	10	3.5	15,000	/	TG/ /	mem
	3.2	6	3.5	22,000	/	/ /	mem
	3.2	10	3.5	37,000	JT/	TG/ /	mem
	10.2	150	3.5	1,767,000	/	TG/ /	MEM
	20.0	170	3.5	3,927,000	/	TG/ /	mem
	28.0	130	3.5	4,204,000	/	TG/ /	mem
	2.0	40	3.5	92,000	/	/ /	mem
	1.0	15	3.5	17,000	/	/ /	mem
Pulau Pinang	3.2	25	3.5	92,000	/	TG/ /	mem
	14.0	130	3.5	2,102,000	/	/ /	mem
	8.3	300	3.5	2,876,000	/	/ /	mem
	9.0	200	1.5	2,079,000	/	/ /	mem
	10.0	250	3.5	2,888,000	/	/ /	mem

THE NATIONAL RIVER MOUTHS STUDY  
IN MALAYSIA

SAMPLE OF BROWSE SCREEN EDITING

Fig. 11.4-15 (1/3)

JAPAN INTERNATIONAL COOPERATION AGENCY

DREDGE	FIS_S_NO1	COM_S_NO	FIS_S_NO2	FIS_S_NO3	FIS_S_NO4	FIS_S_NO5	FIS_S_NO6	FI
MEMO		20	62	89	96	58	51	
memo			32	43	0	0	25	
memo	42		3	175	11	3	3	
memo			25	28	10			
MEMO	1	77	82	72	82	80	104	
memo	2		86	63	2	1		
memo				3				
memo			40	4	0			
MEMO			28	169	2	1	1	
memo		16						
memo			72	163	1	1		
memo	2		139	41				
memo			50	14		5	45	
memo	1		28	452	5			
memo	19		23	174				
memo				39				
memo	8		4	337	362	21	11	
memo	1		3	1				

FIS_S_NO7	FIS_S_NO	FIS_S_NO_E	FISHMEN	FISHMEN_E	FIS_S_SIZE	LANDUSE	CATEGORY	D
76	432	N	2,333	Y	L	UR/VI/AG	1	1
4	104	N	561	Y	L	VI/AG/	1	
1	238	N	762	Y	L	VI/AG/	1	
	63	N	202	Y	M	VI/AG/	1	
115	536	N	1,716	Y	L	UR/VI/AG	1	1
	154	N	493	Y	M	VI/AG/	1	
	3	N	10	Y	M	VI/AG/	3	
	44	N	141	Y	M	VI/AG/	1	
	201	N	504	Y	M	VI/SW/AG	1	
	26	Y	50	N	M	UR/SW/AG	3	
8	245	N	693	N	L	VI/SW/AG	1	
	182	N	700	N	M	UR/VI/AG	1	
8	122	N	50	N	L	VI/AG/	2	
	486	N	1,042	N	M	VI/AG/SW	1	
	216	N	308	N	M	VI/SW/AG	1	
	39	N	76	N	M	VI/SW/	2	
9	752	N	140	N	L	VI/SW/	2	
	5	N	166	N	M	VI/SW/	2	

THE NATIONAL RIVER MOUTHS STUDY  
IN MALAYSIA

JAPAN INTERNATIONAL COOPERATION AGENCY

SAMPLE OF BROWSE SCREEN EDITING

Fig. 11.4-15 (2/3)

FISHMEN_E	FIS_S_SIZE	LANDUSE	CATEGORY	D_BOATSIZE	NPV_COST	NPV_BENEFI
Y	L	UR/VI/AG	1	150	RM49,395,000	RM53,581,000
Y	L	VI/AG/	1	40	RM 8,750,000	RM 1,312,000
Y	L	VI/AG/	1	40	RM 2,394,000	RM 291,000
Y	M	VI/AG/	1	40	RM 1,796,000	RM 53,000
Y	L	UR/VI/AG	1	150	RM44,222,000	RM47,261,000
Y	M	VI/AG/	1	40	RM13,067,000	RM 530,000
Y	M	VI/AG/	3	0	RM 000	RM 000
Y	M	VI/AG/	1	40	RM13,117,000	RM 159,000
Y	M	VI/SW/AG	1	40	RM 6,963,000	RM 106,000
N	M	UR/SW/AG	3	0	RM 0,000	RM 000
N	L	VI/SW/AG	1	40	RM 2,660,000	RM 132,000
N	M	UR/VI/AG	1	40	RM11,391,000	RM 1,086,000
N	L	VI/AG/	2	40	RM11,243,000	RM 1,484,000
N	M	VI/AG/SW	1	40	RM10,738,000	RM 3,193,000
N	M	VI/SW/AG	1	70	RM21,728,000	RM 331,000
N	M	VI/SW/	2	40	RM 5,241,000	RM 0,000
N	L	VI/SW/	2	40	RM 9,667,000	RM 199,000
N	M	VI/SW/	2	40	RM 111,000	RM 0,000

THE NATIONAL RIVER MOUTHS STUDY  
IN MALAYSIA

JAPAN INTERNATIONAL COOPERATION AGENCY

SAMPLE OF BROWSE SCREEN EDITING

Fig. 11.4-15 (3/3)





Process	Card view	Browse view	Maintenance	End
---------	-----------	-------------	-------------	-----

Serial No. (or '?' for name search):

Position with: → - <Enter> to select choice - <F1> Help

Process	Card view	Browse view	Maintenance	End
---------	-----------	-------------	-------------	-----

Serial No. (or '?' for name search): ?  
 Input Name: M ('Be' can be for 'Beruas')

Position with: → - <Enter> to select choice - <F1> Help

Process	Card view	Browse view	Maintenance	End
---------	-----------	-------------	-------------	-----

Serial No. (or '?' for name search): ?  
 Input Name: M ('Be' can be for 'Beruas')

- 61 Marang, Terengganu
- 7 Melaka, Kedah
- 32 Melaka, Melaka
- 63 Merang, Terengganu
- 60 Mercang, Terengganu
- 49 Merchong, Pahang
- 35 Merlimau, Melaka
- 45 Mersing, Johor
- 36 Muar, Johor
- 9 Muda, P.Pinang
- 81 Mukah, Sarawak

Position with: → - <Enter> to select choice - <F1> Help

Process	Card view	Browse view	Maintenance	End
---------	-----------	-------------	-------------	-----

- River mouth name and state
- Map/photo data
- River features
- River mouth features...
- Navigation and fishery
- Socioeconomic condition...
- Present problems
- Researcher's comments
- Categorization and countermeasure

Selected River Mouth: 61. Marang, Terengganu

Position: →↑ Select: ←↓ Help: F1

Basic Information (1/1)		61. Marang Terengganu
Serial No.	:	61
Name	:	Marang
State	:	Terengganu
Division (only for Sarawak):	:	-
District	:	Marang
<Shift>+<PrtSc> Hard copy   <Esc> Prev menu		

THE NATIONAL RIVER MOUTHS STUDY  
IN MALAYSIA

JAPAN INTERNATIONAL COOPERATION AGENCY

SAMPLE HARD COPY OF [River mouth name  
and state] FOR PROCEDURE [Card view]

Fig. 11.4-22

MAP, PHOTO AND CHART (Page 1 of 9)		61. Marang Terengganu
Number of Available Data: 9		
Kind of Data	: C (M:Map P:Aerial Photo C:Navigation Chart)	
Area	: Peninsula (Peninsula or Sarawak or Sabah)	
Series	:	
Scale	:	
Flight Line No.:		
Sheet No.	: Outer	
CEB No.	:	
Date	: 02/ /89 (Month/Date/Year)	
Time	: :	
Source	: DID (LSD:Land & Survey Department) (SMD:Survey & Mapping Department) (MD :Marine Department) (DID:Drainage & Irrigation Department) (ADM:Admiralty Charts)	
<PgDn>/<PgUp> Next/Prev   <Shift>+<PrtSc> Hard copy   <Esc> Prev menu		

MAP, PHOTO AND CHART (Page 2 of 9)		61. Marang Terengganu
Number of Available Data: 9		
Kind of Data	: C (M:Map P:Aerial Photo C:Navigation Chart)	
Area	: Peninsula (Peninsula or Sarawak or Sabah)	
Series	:	
Scale	: 1/ 200,000	
Flight Line No.:		
Sheet No.	: 770	
CEB No.	:	
Date	: 04/29/88 (Month/Date/Year)	
Time	: :	
Source	: ADM (LSD:Land & Survey Department) (SMD:Survey & Mapping Department) (MD :Marine Department) (DID:Drainage & Irrigation Department) (ADM:Admiralty Charts)	
<PgDn>/<PgUp> Next/Prev   <Shift>+<PrtSc> Hard copy   <Esc> Prev menu		

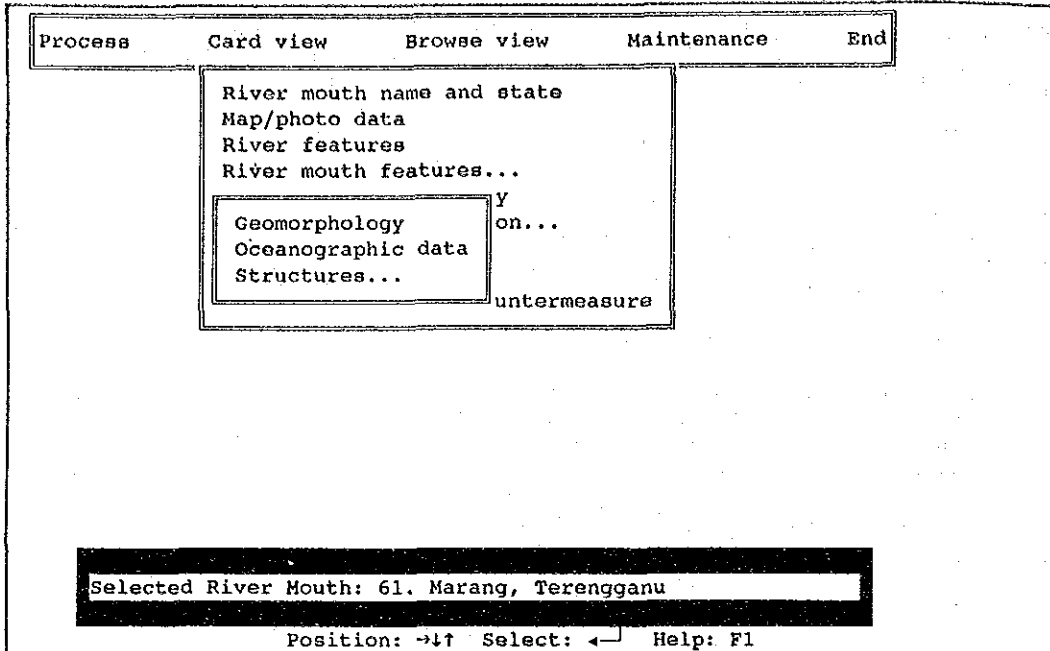
MAP, PHOTO AND CHART (Page 4 of 9)	61. Marang Terengganu
Number of Available Data: 9	
Kind of Data : M (M:Map P:Aerial Photo C:Navigation Chart)	
Area	: Peninsula (Peninsula or Sarawak or Sabah)
Series	: L 7010
Scale	: 1/ 63,360
Flight Line No.:	
Sheet No.	: 38
CEB No.	:
Date	: / / (Month/Date/Year)
Time	: :
Source	: SMD (LSD:Land & Survey Department) (SMD:Survey & Mapping Department) (MD :Marine Department) (DID:Drainage & Irrigation Department) (ADM:Admiralty Charts)
<PgDn>/<PgUp> Next/Prev   <Shift>+<PrtSc> Hard copy   <Esc> Prev menu	

MAP, PHOTO AND CHART (Page 5 of 9)	61. Marang Terengganu
Number of Available Data: 9	
Kind of Data : M (M:Map P:Aerial Photo C:Navigation Chart)	
Area	: Peninsula (Peninsula or Sarawak or Sabah)
Series	: L 7030
Scale	: 1/ 50,000
Flight Line No.:	
Sheet No.	: 4365
CEB No.	:
Date	: / / (Month/Date/Year)
Time	: :
Source	: SMD (LSD:Land & Survey Department) (SMD:Survey & Mapping Department) (MD :Marine Department) (DID:Drainage & Irrigation Department) (ADM:Admiralty Charts)
<PgDn>/<PgUp> Next/Prev   <Shift>+<PrtSc> Hard copy   <Esc> Prev menu	

MAP, PHOTO AND CHART (Page 6 of 9)	61. Marang Terengganu
Number of Available Data: 9	
Kind of Data : P (M:Map P:Aerial Photo C:Navigation Chart)	
Area	: Peninsula (Peninsula or Sarawak or Sabah)
Series	: 1966
Scale	: 1/ 25,000
Flight Line No.:	C46 L79 S
Sheet No.	:
CEB No.	: 25 - 66 EC
Date	: / /66 (Month/Date/Year)
Time	: 14:25
Source	: SMD (LSD:Land & Survey Department) (SMD:Survey & Mapping Department) (MD :Marine Department) (DID:Drainage & Irrigation Department) (ADM:Admiralty Charts)
<PgDn>/<PgUp> Next/Prev   <Shift>+<PrtSc> Hard copy   <Esc> Prev menu	

MAP, PHOTO AND CHART (Page 8 of 9)	61. Marang Terengganu
Number of Available Data: 9	
Kind of Data : P (M:Map P:Aerial Photo C:Navigation Chart)	
Area	: Peninsula (Peninsula or Sarawak or Sabah)
Series	: 1974
Scale	: 1/ 25,000
Flight Line No.:	F55 L99 N
Sheet No.	:
CEB No.	: 9 - 74 EC
Date	: 06/ /74 (Month/Date/Year)
Time	: 08:26
Source	: SMD (LSD:Land & Survey Department) (SMD:Survey & Mapping Department) (MD :Marine Department) (DID:Drainage & Irrigation Department) (ADM:Admiralty Charts)
<PgDn>/<PgUp> Next/Prev   <Shift>+<PrtSc> Hard copy   <Esc> Prev menu	

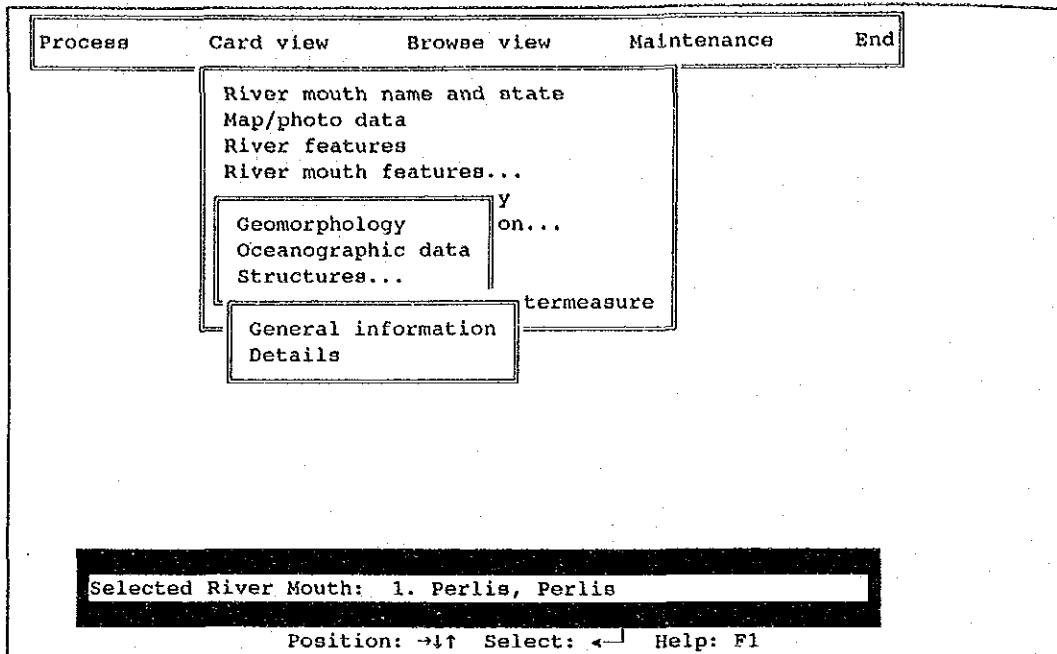
River Features (1/1)	61. Marang Terengganu
Catchment Area:	460 km <sup>2</sup>
River Length :	44.4 km
Riverbed Gradient at the River Mouth:	1/
Flow Capacity in the Downstream Area:	m <sup>3</sup> /s
Average Annual Surface Soil Loss :	421 ton/km <sup>2</sup> /year
Water Quality (Observed on 02/18/92)	
Turbidity :	LW (HG:High MD:Medium LW:Low CL:Clear OT:Others)
Color :	GR (GY:Gray BR:Brown BL:Black GR:Green OT:Others)
Odor :	NO (HG:High MD:Medium LW:Low NO:No Smell)
Temperature:	29 centi grade
Availability of Data in FORM-1 (Y/N)	
Rainfall: N	Water Level: N Discharge: N Sediment: N
Water Quality: N	
<Shift>+<PrtSc> Hard copy   <Esc> Prev menu	





River Mouth Geomorphological Features (1/1)		61. Marang Terengganu	
Location of River Mouth: OB (IB:Inner Bay    OB:Outside Bay)			
Condition of Beach Change : FR		VS:Very Serious    SR:Serious	
Shoreline Change: FR		FR: Fair    NP:No Problem	
River Mouth Form and Dimensions			
Form: ON	OW:Open Wide    ON:Open Narrow    ST:Straight CL:Closed Narrow    CD:Closed		
Width: 244.0 m	Depth: 1.6 m	Assmd. Min. Depth: 0.5 m	
Bed Material			
Seabed : FS	CL:Clay	ST:Silt	SS:Silt & Sand
Littoral Area: MS/	VS:Very fine sand	FS:Fine Sand	
River Mouth : SS/	MS:Medium Sand	CS:Coarse Sand	
Riverbed : SS	GR:Gravel	CB:Cobble	BD:Boulder
<Shift>+<PrtSc> Hard copy   <Esc> Prev menu			

River Mouth Oceanographical Features (1/1)		61. Marang Terengganu	
Wave Class: WS (WS:High Straight Wave    WO:High Oblique Wave    LW:Low Wave)			
Nearest Standard Tidal Port: Cendering			
Tidal Prism			
Tidal Intrusion Stretch	:	12.0 km	
Average Width of River	:	80 m	
Tidal Range (Astronomical Max. Range):	:	3.5 m	
Tidal Prism Volume	:	1,109,000 m <sup>3</sup>	
<Shift>+<PrtSc> Hard copy   <Esc> Prev menu			



Structures at the River Mouth General Information (1/2)	1. Perlis Perlis
Structures for River Mouth Protection	
Existence of Structures (Y/N): N	
Purpose of the Structure: / / /	HT:High Tide HW:High Water HL:High Water Level FL:Flush Out RM:Stabilize River Mouth OT:Others
Present Working Condition :	VG:Very Good    GD:Good FR:Fair        PR:Poor
Present Structural Condition:	VP:Very Poor
Availability of Data in FORM-6 (Y/N): N	
<PgDn>/<PgUp> Next/Prev   <Shift>+<PrtSc> Hard copy   <Esc> Prev menu	

Structures at the River Mouth General Information (2/2)	1. Perlis Perlis
Information on Dredging	
'86, '87, '90, '91, (Schd'92) by MD	
	<F9>/<Esc> Zoom/Return
Hydraulic Structures	
Type of Structures: / / / / /	TG:Tidal Gate    EB:Earth Bund    JT:Jetty IP:Intake Pumphouse    DP:Drainage Pumphouse OG:Outlet Gate    OT:Others
Availability of Data in FORM-6 (Y/N): N	
<PgDn>/<PgUp> Next/Prev   <Shift>+<PrtSc> Hard copy   <Esc> Prev menu	

Structures at the River Mouth Detailed Information (1/2)		14. Tg. Piandang Perak
Category: ST (ST:Hydraulic Structures    PW:Protection Works)		
Type : TG	TG:Tidal Gate    EB:Earth Bund JT:Jetty    IP:Intake Pumphouse DP:Drainage Pumphouse OG:Outlet Gate    OT:Others	
Purpose : Tidal / drainage		
Location: 400m on left side of river		
Size		
Dimension:		
Height: 4.0 m	Width: 1.2 m	Length: 8.0m
<PgDn>/<PgUp> Nest/Prev   <Shift>+<PrtSc> Hard copy   <Esc> Prev menu		

Structures at the River Mouth Detailed Information (2/2)		14. Tg. Piandang Perak
Construction		
Date Started: / /	Date Completed: / /91	
Cost: RM		
Managing Agency		
Section :		
Department: DID		
City : Bagan Serai		
Memo	<div style="border: 1px solid black; height: 40px; width: 100%;"></div>	
<F9>/<Esc> Zoom/Return		
<PgDn>/<PgUp> Nest/Prev   <Shift>+<PrtSc> Hard copy   <Esc> Prev menu		

Navigation and Fishery (1/1)						62. Terengganu Terengganu	
Number of Commercial Boat: 161							
Number of Fishing Boat							
Non- Powered	Outboard Engine	=====	Inboard Engine (GRT)=====			Total	
		10.9& below	11- 20.9	21- 30.9	31- 40.9	41 & above	
		3	35	32	7	10	20
<----Small---->		<-----Medium----->			<Large>		Estimated? (Y/N): N
Typical Boat Size: L (S:Small M:Medium L:Large)							
Number of Fishermen: 417 Is this estimated Value? (Y/N): Y							
Location of Fishing: RM/ /							
<div style="border: 1px solid black; padding: 2px; display: inline-block;">           RM:around River Mouth            IN:Inshore OF:Offshore            OT:Others         </div>							
<Shift>+<PrtSc> Hard copy   <Esc> Prev menu							

Process	Card view	Browse view	Maintenance	End
---------	-----------	-------------	-------------	-----

<ul style="list-style-type: none"> <li>River mouth name and state</li> <li>Map/photo data</li> <li>River features</li> <li>River mouth features...</li> <li>Navigation and fishery</li> <li>Socioeconomic condition...</li> </ul>
---

<ul style="list-style-type: none"> <li>Water use</li> <li>Land use, vegetation &amp; urban centers</li> <li>Development plans</li> <li>Fauna/flora</li> <li>Activities and others</li> </ul>
--

Selected River Mouth: 19. Beruas, Perak

Position: →↑ Select: ← Help: F1

Water Use (1/1)		19. Beruas Perak						
River Water Utilization: AG/DM/FS/								
<table border="1"> <tr> <td>ID:Industry</td> <td>AG:Agriculture</td> </tr> <tr> <td>DM:Domestic Water</td> <td>FS:Fish Pond</td> </tr> <tr> <td>AQ:Aquaculture</td> <td>OT:Others</td> </tr> </table>			ID:Industry	AG:Agriculture	DM:Domestic Water	FS:Fish Pond	AQ:Aquaculture	OT:Others
ID:Industry	AG:Agriculture							
DM:Domestic Water	FS:Fish Pond							
AQ:Aquaculture	OT:Others							
Description:	<table border="1"> <tr> <td>dam - for hydroelectric power (small)</td> </tr> <tr> <td>- water supply</td> </tr> </table>		dam - for hydroelectric power (small)	- water supply				
dam - for hydroelectric power (small)								
- water supply								
<F9>/<Esc> Zoom/Return <Shift>+<PrtSc> Hard copy   <Esc> Prev menu								

Land Use, Vegetation and Urban Centers (1/2) 19. Beruas Perak

Present Land Use near the Mouth: UR/SW/AG

UR:Urban Area VI:Village AG:Agriculture Land  
FO:Forest SW:Swamp UN:Unused Land

Publishment for Land Use

Year : 79-  
Title : The Present Landuse of Peninsular Malaysia  
Page No. :

Managing Agency  
Section :  
Department : Ministry of Agriculture  
City :

Memo for Land Use

<F9>/<Esc> Zoom/Return  
<PgDn>/<PgUp> Next/Prev | <Shift>+<PrtSc> Hard copy | <Esc> Prev. menu

Land Use, Vegetation and Urban Centers (2/2) 19. Beruas Perak

Major Urban Centers

City Name : Beruas

Location :

Distance from  
the River Mouth: 25 km

Population : 3,876

Dependence Degree: %

<PgDn>/<PgUp> Next/Prev | <Shift>+<PrtSc> Hard copy | <Esc> Prev menu



Process	Card view	Browse view	Maintenance	End
	River mouth name and state			
	Map/photo data			
	River features			
	River mouth features...			
	Navigatio			
	Socioecon	No objective information.		
	Water us			
	Land use, vegetation & urban centers			
	Development plans			
	Fauna/flora			
	Activities and others			
Selected River Mouth: 19. Beruas, Perak				

THE NATIONAL RIVER MOUTHS STUDY  
IN MALAYSIA

JAPAN INTERNATIONAL COOPERATION AGENCY

SAMPLE OF MESSAGE FOR NO OBJECTIVE  
INFORMATION

Fig. 11.4-35

Development Plans (1/2)		1. Perlis Perlis
Existence of Development Plans (Y/N): Y		
Development Plans Information		
Name	:	RTB I
Objective	:	Kangar Flood Mitigation
Implementation		
Started	:	/ /86
Ended	:	/ /88
Period	:	
Work Volume :		
Cost	:	RM2,799,430.-
<PgDn>/<PgUp> Next/Prev   <Shift>+<PrtSc> Hard copy   <Esc> Prev menu		

Development Plans (2/2)		1. Perlis Perlis
Managing Office		
Section	:	
Department	:	DID
City	:	Perlis
Address	:	04-761380
Memo for Development Plans		
<div style="border: 1px solid black; padding: 5px;">           RTB I : Flood Mitigation Scheme Phase I (Digging Perlis River)         </div>		
<F9>/<Esc> Zoom/Return		
<PgDn>/<PgUp> Next/Prev   <Shift>+<PrtSc> Hard copy   <Esc> Prev menu		

Flora and Fauna (1/1)	69. Sematan Sarawak
<p>Publication/Data of Flora and Fauna</p> <p>Year : 80-</p> <p>Title : A Checklist of the trees of Sarawak, J.A.R.Anderso</p> <p>Page No. :</p> <p>Managing Agency</p> <p>Section : National Park &amp; Wildlife Section</p> <p>Department : Forest Department</p> <p>City : Sarawak</p> <p>Memo for Flora and Fauna</p> <div style="border: 1px solid black; height: 50px; width: 100%; margin-top: 10px;"></div>	
<p>&lt;F9&gt;/&lt;Esc&gt; Zoom/Return</p> <p>&lt;PgDn&gt;/&lt;PgUp&gt; Next/Prev   &lt;Shift&gt;+&lt;PrtSc&gt; Hard copy   &lt;Esc&gt; Prev menu</p>	

Activities at the River Mouth and Others (1/1)	1. Perlis Perlis
4 pages	
<p>Existence of Activity (Y/N): Y</p> <p>Kind of Activity: FS/RF/RC</p> <div style="border: 1px solid black; padding: 5px; margin-top: 10px;"> <p>RC:Recreational Activities (fishing, boating, swimming, etc.)</p> <p>FS:Fishery    DP:Water Disposal    MI:Mining (sand)</p> <p>RF:Recreational Facilities    AG:Agriculture</p> <p>TD:Trading    FO:Forestry    OT:Others</p> </div> <p>Local Interviewees' Memo for Socioeconomic Condition</p> <p>Interviewee No.: 1/4</p> <div style="border: 1px solid black; padding: 5px; margin-top: 10px;"> <p>The Kuala Perlis jetty serves the ferry for Langkawi Island and it attract a lot of tourist especially at weekends and holidays.</p> </div>	
<p>&lt;F9&gt;/&lt;Esc&gt; Zoom/Return</p> <p>&lt;PgDn&gt;/&lt;PgUp&gt; Next/Prev   &lt;Shift&gt;+&lt;PrtSc&gt; Hard copy   &lt;Esc&gt; Prev menu</p>	

Present Problems (1/4)		9: Muda P.Pinang	
River Mouth Clogging Condition			
Cause of Clogging No.1: SB		Cause of Clogging No.2: ST	
SB:Sand Bar    ST:Siltation    OT:Others			
=====		=====	
Present Clogging Condition: SR		Present Clogging Condition:	
VS:Very Serious    SR:Serious FR:Fair    NP:No Problem			
Clogging Period: DA		Clogging Period:	
DA:Daily    SE:Seasonally    YR:Yearly AY: All Year    OT: Other			
<PgDn>/<PgUp> Next/Prev   <Shift>+<PrtSc> Hard copy   <Esc> Prev menu			

Present Problems (2/4)		9: Muda P.Pinang	
River Mouth Clogging Condition			
Cause of Clogging No.1: SB		Cause of Clogging No.2: ST	
SB:Sand Bar    ST:Siltation    OT:Others			
=====		=====	
Annual Clogging Frequency: 7V		Annual Clogging Frequency:	
Times/Year 00:0-1    12:1-2    34:3-4    56:5-6    7V:7over    AY:All Year			
Location of Clogging: RM/ /		Location of Clogging: / /	
BE:Before River Mouth    RM:around River Mouth OF:offshore    OT:Others			
<PgDn>/<PgUp> Next/Prev   <Shift>+<PrtSc> Hard copy   <Esc> Prev menu			

Present Problems (3/4)	9: Muda P.Pinang						
River Mouth Clogging Condition							
Classification of Damage Caused by Clogging: OT/NV/ /							
<table border="1"> <tr> <td>NV:Navigation</td> <td>WS:Water Supply</td> <td>PL:Pollution</td> </tr> <tr> <td>FS:Fish Swim Up</td> <td>OT:Others</td> <td>NP:No Problem</td> </tr> </table>		NV:Navigation	WS:Water Supply	PL:Pollution	FS:Fish Swim Up	OT:Others	NP:No Problem
NV:Navigation	WS:Water Supply	PL:Pollution					
FS:Fish Swim Up	OT:Others	NP:No Problem					
Riverbed Rise caused the difficulty of: NV/ /							
<table border="1"> <tr> <td>DR:Drainage</td> <td>FL:Flood</td> <td>NV:Navigation</td> </tr> <tr> <td>OT:Others</td> <td>NP:No Problem</td> <td></td> </tr> </table>		DR:Drainage	FL:Flood	NV:Navigation	OT:Others	NP:No Problem	
DR:Drainage	FL:Flood	NV:Navigation					
OT:Others	NP:No Problem						
Cause of Navigation Difficulty: DP/ /							
<table border="1"> <tr> <td>DP:Not Enough Depth</td> <td>WD:Width</td> <td>SL:Ship Line</td> </tr> <tr> <td>WP:Water Pollution</td> <td>OT:Others</td> <td>NP:No Problem</td> </tr> </table>		DP:Not Enough Depth	WD:Width	SL:Ship Line	WP:Water Pollution	OT:Others	NP:No Problem
DP:Not Enough Depth	WD:Width	SL:Ship Line					
WP:Water Pollution	OT:Others	NP:No Problem					
<PgDn>/<PgUp> Next/Prev   <Shift>+<PrtSc> Hard copy   <Esc> Prev menu							

Present Problems (4/4)	9: Muda P.Pinang								
Existence of Commercial Navigation Difficulty (Y/N): N									
Complaint from Fishermen: 2 (0:No 1:Some 2:Strong)									
Existence of Flooding Problem: N									
Water Supply Difficulty in the Sector of: / /									
<table border="1"> <tr> <td>ID:Industry</td> <td>AG:Agriculture</td> <td>PL:Plantation</td> <td>AQ:Aquaculture</td> </tr> <tr> <td>FS:Fishpond</td> <td>DM:Domestic Water</td> <td>OT:Others</td> <td></td> </tr> </table>		ID:Industry	AG:Agriculture	PL:Plantation	AQ:Aquaculture	FS:Fishpond	DM:Domestic Water	OT:Others	
ID:Industry	AG:Agriculture	PL:Plantation	AQ:Aquaculture						
FS:Fishpond	DM:Domestic Water	OT:Others							
Water Pollution Source: /									
<table border="1"> <tr> <td>OP:Oil Palm Effluent</td> <td>AG:Agriculture</td> <td>OT:Others</td> </tr> </table>		OP:Oil Palm Effluent	AG:Agriculture	OT:Others					
OP:Oil Palm Effluent	AG:Agriculture	OT:Others							
<PgDn>/<PgUp> Next/Prev   <Shift>+<PrtSc> Hard copy   <Esc> Prev menu									

Socioeconomic Effects of the Project

The river is the state boundary line between Kedah and Penang. Therefore, the river mouth is crowded with fishing boats from Penang and Kedah. The main activity here is fishing with facilities such as fish trading centre, and food stalls right at the river mouth. There are fishing villages at both sides of the river mouth. From our observation, navigation through the river mouth is a problem even for a small size boat due to the sand bars and siltation.

<F9>/<Esc> Zoom/Return  
<PgDn>/<PgUp> Next/Prev | <Shift>+<PrtSc> Hard copy | <Esc> Prev menu

Applicable Countermeasures

Dredging at regular basis is recommended to improve the river mouth channel. However as the river mouth tends to be quickly clogged up by sand bars, a permanent solution to the problem should be sought by carrying a detailed study.

<F9>/<Esc> Zoom/Return  
<PgDn>/<PgUp> Next/Prev | <Shift>+<PrtSc> Hard copy | <Esc> Prev menu

Researcher's Comment

1. The clogging problem at the river mouth is due to mainly by sandbars and also by siltation. It was reported that the river mouth was clogged up again by sandbars within one year after dredging work was completed in 1986.
2. Besides the clogging problem, the river mouth is also experiencing coastal erosion as indicated by rock gabion wall at the right side of the river mouth.
3. The water resources of this river is becoming increasingly important to both states. As it is Kedah (MADA) is operating Muda Dam, located in its upper reaches, to transfer water through the Saiong Tunnel into Pedu Dam in Sungai Kedah for irrigation and water supply. Along the way water is also extracted for irrigation and water supply in both states. A proposal was also made to construct additional dams, namely Reman, Naok and Beris which will then release the water into

<F9>/<Esc> Zoom/Return

<PgDn>/<PgUp> Next/Prev | <Shift>+<PrtSc> Hard copy | <Esc> Prev menu

- [ .....v1.....v2.....v3.....v4.....v5.....v6.....].....7.....
1. The clogging problem at the river mouth is due to mainly by sandbars and also by siltation. It was reported that the river mouth was clogged up again by sandbars within one year after dredging work was completed in 1986.
  2. Besides the clogging problem, the river mouth is also experiencing coastal erosion as indicated by rock gabion wall at the right side of the river mouth.
  3. The water resources of this river is becoming increasingly important to both states. As it is Kedah (MADA) is operating Muda Dam, located in its upper reaches, to transfer water through the Saiong Tunnel into Pedu Dam in Sungai Kedah for irrigation and water supply. Along the way water is also extracted for irrigation and water supply in both states. A proposal was also made to construct additional dams, namely Reman, Naok and Beris which will then release the water into the river. Some of this will be diverted into the Muda Irrigation area for irrigation by ways of Jeniang Transfer Canal. It is understood that Beris Dam will be implemented soon.
  4. Sand extraction is carried out some 10 km upstream for use in the construction industry.

Categorization and Countermeasure (1/1)

61. Marang  
Terengganu

Category : 1  
(1:Critical 2:Significant 3:Acceptible)

Design Boat Size : 40 GRT

Project Implementation (Master Plan)

Net Present Value of Cost : RM17,335,000.-

Net Present Value of Benefit: RM 3,670,000.-

<Shift>+<PrtSc> Hard copy | <Esc> Prev menu

THE NATIONAL RIVER MOUTHS STUDY  
IN MALAYSIA

JAPAN INTERNATIONAL COOPERATION AGENCY

SAMPLE HARD COPY OF [Categorization and  
countermeasure] FOR PROCEDURE [Card  
view]

Fig. 11.4-42



Process	Card view	Browse view	Maintenance	End
---------	-----------	-------------	-------------	-----

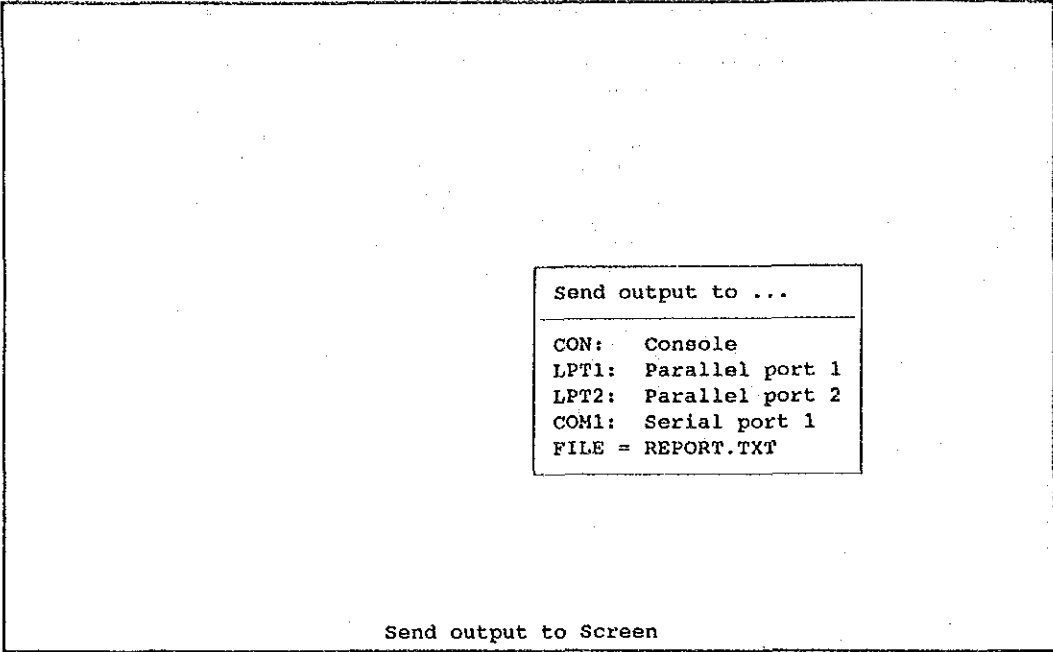
River mouth list  
River features  
River mouth features...  
Navigation and fishery  
Socioeconomic conditions...  
Present problems  
Categorization and countermeasure

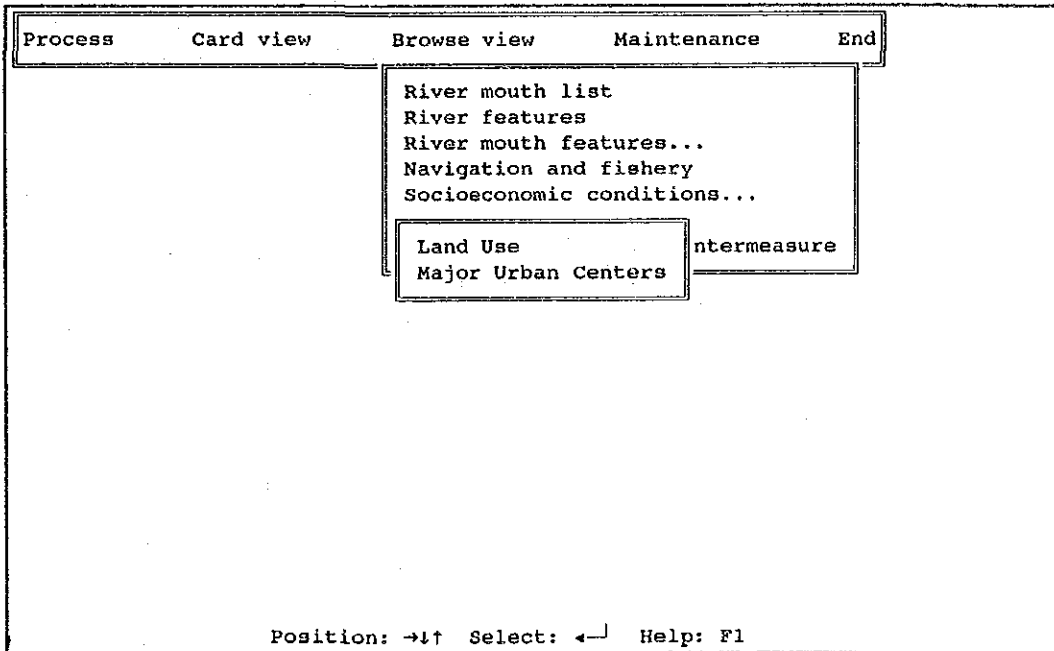
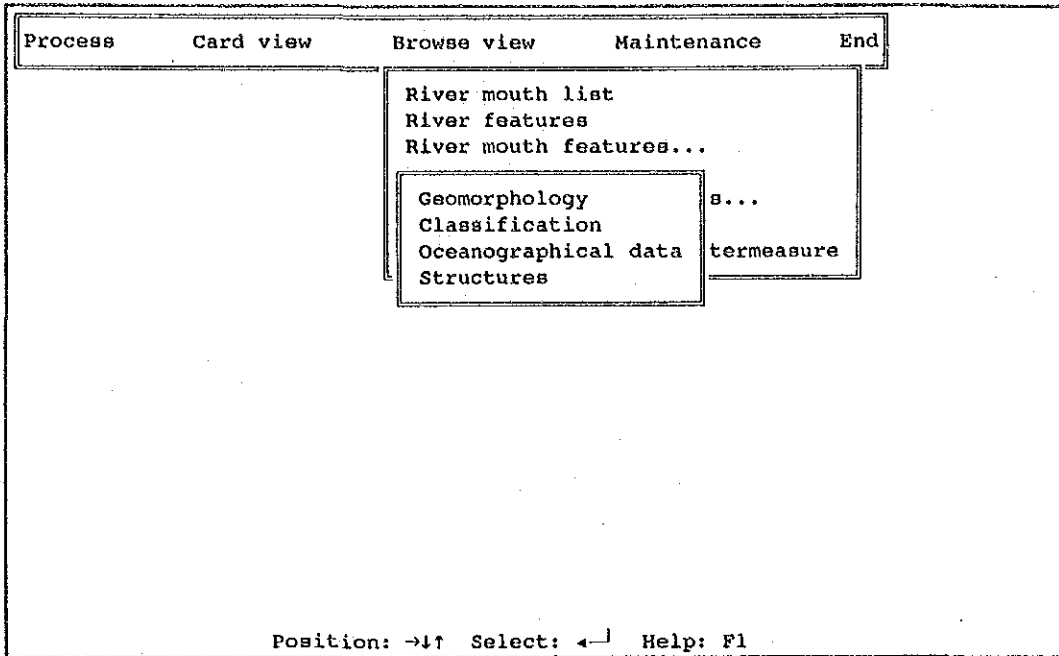
Position: →↑ Select: ←↓ Help: F1

Process	Card view	Browse view	Maintenance	End
---------	-----------	-------------	-------------	-----

River mouth list  
River features  
River mouth features...  
Navigation and fishery

Printer ready? (Y/N): Y





Process	Card view	Browse view	Maintenance	End
---------	-----------	-------------	-------------	-----

Change password  
 Back up file to floppy  
 Restore file from floppy

Position: →↑ Select: ← Help: F1

Process	Card view	Browse view	Maintenance	End
---------	-----------	-------------	-------------	-----

Change password  
 Back up file to floppy  
 Restore file from floppy

Enter Old Password:

THE NATIONAL RIVER MOUTHS STUDY IN MALAYSIA	SUB-MENU OF PROCEDURE [Maintenance]	Fig. 11.4-48
JAPAN INTERNATIONAL COOPERATION AGENCY	PASSWORD INPUT PROMPT FOR [Change password] FOR PROCEDURE [Maintenance]	Fig. 11.4-49

Process	Card view	Browse view	Maintenance	End
---------	-----------	-------------	-------------	-----

Change password  
 Back up file to floppy  
 Restore file from floppy

Access denied!  
 (Password is case(uppercase/lowercase) sensitive.)

Process	Card view	Browse view	Maintenance	End
---------	-----------	-------------	-------------	-----

Change password  
 Back up file to floppy  
 Restore file from floppy

Enter New Password:

THE NATIONAL RIVER MOUTHS STUDY  
IN MALAYSIA

JAPAN INTERNATIONAL COOPERATION AGENCY

ACCESS DENIED MESSAGE FOR PASSWORD  
CONTROL FOR PROCEDURE [Maintenance]

NEW PASSWORD INPUT PROMPT FOR [Change  
password] FOR PROCEDURE [Maintenance]

Fig. 11.4-50

Fig. 11.4-51

Process	Card view	Browse view	Maintenance	End
---------	-----------	-------------	-------------	-----

Change password  
 Back up file to floppy  
 Restore file from floppy

Are you sure to backup files? (Y/N): N

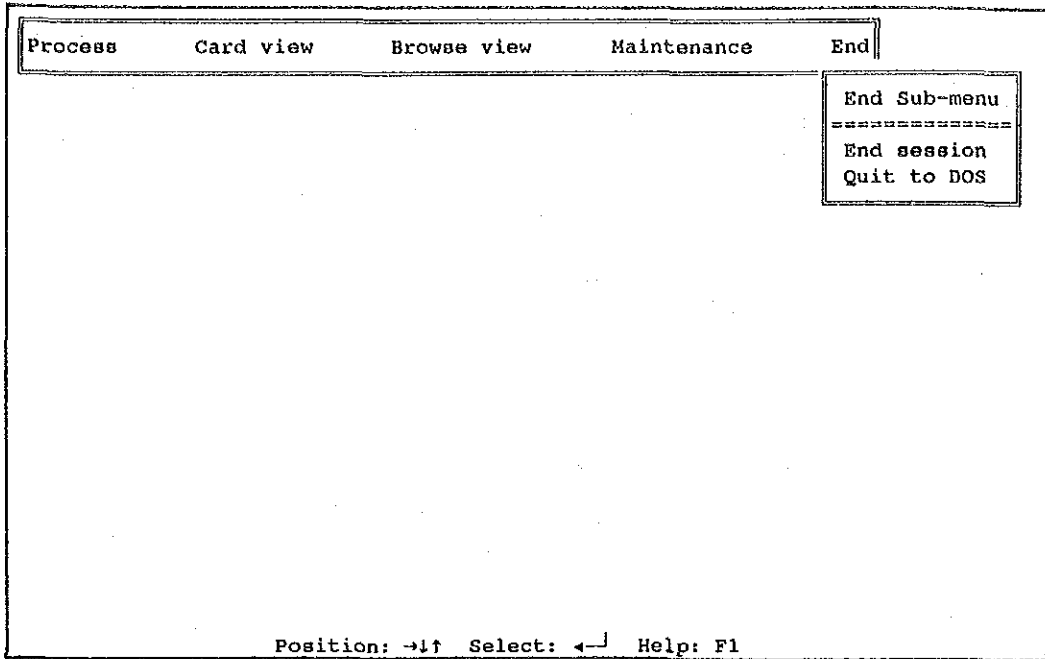
Position: →↑ Select: ←␣ Help: F1

Process	Card view	Browse view	Maintenance	End
---------	-----------	-------------	-------------	-----

Change password  
 Back up file to floppy  
 Restore file from floppy

Are you sure to restore files? (Y/N): N

Position: →↑ Select: ←␣ Help: F1







*ANNEXES*



```

PROCEDURE R_M_BAT
PARAMETER entryflg
*-----
* Program.....: R_M_BAT.PRG
* Author.....: K.Sasabe, National River Mouths Study
* Date.....: 8-27-93
* Notice.....: Japan International Cooperation Agency (JICA)
* Generated by.: dBASE IV version 1.5
* Description..: Menu actions
*-----
PRIVATE gc_prognum
gc_prognum="01"
SET COLOR OF NORMAL TO W+/B
CLEAR
PRIVATE lc_ApGen
lc_ApGen=LTRIM(STR(gn_ApGen))

DO SET01
IF gn_error > 0
  gn_error=0
  RETURN
ENDIF

lc_popmenu="R_M_BAT"
DO ACT01

RETURN
*-- EOP: R_M_BAT

PROCEDURE SET01
*-----
* R_M_BAT setup routine. This procedure will open up the file or
* view for this menu, and setup the color scheme.
*-----
ON KEY LABEL F1 DO 1HELP1          && Define F1-Help key
RETURN
*-- EOP: SET01 - Menu R_M_BAT

PROCEDURE ACT01
*-----
* Begin R_M_BAT: BATCH Menu Actions.
* Includes Before item, action, and after item as needed
*-----
PRIVATE lc_new, lc_dbf
lc_new=' '
lc_dbf=' '
listval=PROMPT()
PRIVATE gl_batch
gl_batch=.T.

*-- Action #1 Initial Settings
SET MESSAGE TO
*-- Desc: Inline DO dBASE commands
SET TALK OFF
SET DELETED ON
SET EXACT ON
SET CONFIRM ON
SET SAFETY OFF
SET STATUS OFF
SET SCOREBOARD OFF
SET PROCEDURE TO RMPROC
SELECT 1
USE RESEAR 1 ORDER SERIAL
DEFINE WINDOW MAIN FROM 0,0 TO 20,79 NONE
DEFINE WINDOW ALARM FROM 7,25 TO 10,78 DOUBLE COLOR ,,R*
DEFINE WINDOW PASS FROM 7,25 TO 9,56 DOUBLE COLOR ,N/N
DEFINE WINDOW PRINT FROM 0,0 TO 23,79 NONE
DEFINE WINDOW ASK1 FROM 4,20 TO 7,75 DOUBLE
DEFINE WINDOW ASK2 FROM 8,29 TO 20,55 NONE
DEFINE WINDOW NAME FROM 21,5 TO 23,65 PANEL COLOR ,,G+
DEFINE WINDOW POPUP SCHOICE FROM 0,0 PROMPT FIELD SERIAL
ACTIVATE WINDOW MAIN

*-- Action #2 Printer Setting
SET MESSAGE TO
*-- Desc: Inline DO dBASE commands
ACTIVATE WINDOW alarm
mok="N"
@ 0,0 SAY "Initialize Printer? (Y/N) If Yes, on-line printer." GET mok
READ
DEACTIVATE WINDOW alarm
IF UPPER(mok)="Y"
  SET TALK OFF

```

```

SET PRINT ON
?? CHR(27)+"E"+CHR(27)+"(10U"+CHR(27)+"(sOp12H"+CHR(27)+"&126A"
?? CHR(27)+"&100"+CHR(27)+"&a11L"+CHR(27)+"&14E"
SET PRINT OFF
SET DEVICE TO SCREEN
DEACTIVATE WINDOW main
ACTIVATE WINDOW main
ENDIF

*-- Action #3 Additional Setting
SET MESSAGE TO
*-- Desc: Inline DO dBASE commands

*-- Action #4 Start Main Menu
DO MAIN WITH "B01"

*-- Action #5 End Session
SET MESSAGE TO
*-- Desc: Inline DO dBASE commands
DEACTIVATE WINDOW main
RELEASE WINDOW main
RELEASE ALL
SET DELETED OFF
SET EXACT OFF
SET CONFIRM OFF
SET SAFETY ON
SET STATUS ON
SET SCOREBOARD ON
SET TALK ON

gl_batch=.F.

SET MESSAGE TO
RETURN
*-- EOP: ACT01 - Menu R_M_BAT

PROCEDURE MAIN
PARAMETER entryflg
*-----
* Program.....: MAIN.PRG
* Author.....: K.Sasabe, National River Mouths Study
* Date.....: 8-27-93
* Notice.....: Japan International Cooperation Agency (JICA)
* Generated by.: dBASE IV version 1.5
* Description..: Menu actions
*-----
PRIVATE gc_prognum
gc_prognum="02"

SAVE SCREEN TO SET02&lc_Apgen.
DO SET02
IF gn_error > 0
  gn_error=0
  RETURN
ENDIF

ACTIVATE MENU MAIN

RESTORE SCREEN FROM SET02&lc_Apgen.
RELEASE SCREEN SET02&lc_apgen
RETURN
*-- EOP: MAIN

PROCEDURE SET02
*-----
* MAIN setup routine. This procedure will open up the file or
* view for this menu, and setup the color scheme.
* It will then paint the menu on the screen.
*-----

ON KEY LABEL F1 DO 1HELP1          && Define F1-Help key

IF gn_error = 0
  IF gl_color .AND. .NOT. SET("ATTRIBUTES") = ;
    "W/B,RG+/GB,N/N "+CHR(38)+CHR(38)+" RG+/N,W/B,RG+/GB,B/W,N/GB"
    SET COLOR OF NORMAL TO W+/B
    SET COLOR OF MESSAGES TO RG+/N
    SET COLOR OF TITLES TO W/B
    SET COLOR OF HIGHLIGHT TO RG+/GB
    SET COLOR OF BOX TO RG+/GB
    SET COLOR OF INFORMATION TO B/W
    SET COLOR OF FIELDS TO N/GB
  ENDIF

  DO SET02A                        && Paint the bar menu

```

```

ENDIF
RETURN
*-- EOP: SET02 - Menu MAIN

PROCEDURE SET02A
*-----
* MAIN menu paint routine.
*-----
PRIVATE ll exact
SET BORDER TO
@ 0,1 TO 2,67 DOUBLE COLOR RG+/GB
@ 1,2 CLEAR TO 1,66
@ 1,2 FILL TO 1,66 COLOR RG+/N
@ 1,2 SAY "Process" COLOR RG+/N
@ 1,15 SAY "Card view" COLOR RG+/N
@ 1,30 SAY "Browse view" COLOR RG+/N
@ 1,47 SAY "Maintenance" COLOR RG+/N
@ 1,64 SAY "End" COLOR RG+/N
ll exact = SET("EXACT") = "ON"           && Save Exact state
SET EXACT ON                             && Force Exact on for menu name compare
lc_popup = POPUP()                       && Find out the active pulldown

*-- Determine which menu pad to draw as highlight
IF lc_popup = "PROCESS"
@ 1,2 SAY "Process" COLOR RG+/GB
ENDIF
IF lc_popup = "CARDVIEW"
@ 1,15 SAY "Card view" COLOR RG+/GB
ENDIF
IF lc_popup = "BROWVIEW"
@ 1,30 SAY "Browse view" COLOR RG+/GB
ENDIF
IF lc_popup = "MAINTE"
@ 1,47 SAY "Maintenance" COLOR RG+/GB
ENDIF
IF lc_popup = "END"
@ 1,64 SAY "End" COLOR RG+/GB
ENDIF
IF .NOT. ll exact                         && If Exact was off
SET EXACT OFF                             && Turn it back off
ENDIF

```

```

RETURN
*-- EOP: SET02A - Menu MAIN

```

```

PROCEDURE ACT02
*-----
* Begin MAIN: BAR Menu Actions.
* Includes before item, action, and after item as needed
*-----

```

```

PRIVATE lc_new, lc_dbf
lc_new=' '
lc_dbf=' '
gc_scope=''
DO CASE

*-- Process
CASE "PAD 1" = PAD()
  mhit=.F.
  DO PASSWORD WITH mhit
  IF mhit=.F.
    RETURN
  ENDIF
  ACTIVATE WINDOW main
  lc_new='Y'
  DO PROCESS WITH " 02"

*-- Card view
CASE "PAD 2" = PAD()
  mok="N"
  serialno=0
  mhit=.F.
  mdo=.T.
  DO WHILE mhit=.F.
    DO SERIALCK WITH mhit,serialno,mdo
    IF mdo=.F.
      DEACTIVATE WINDOW name
      ACTIVATE WINDOW main
      RETURN
    ENDIF
  LOOP
ENDDO
CLEAR TYPEAHEAD
ACTIVATE WINDOW main

```

```

lc_new='Y'
DO CARDVIEW WITH " 02"
DEACTIVATE WINDOW name

*-- Browse view
CASE "PAD 3" = PAD()
  lc_new='Y'
  DO BROWVIEW WITH " 02"

*-- Maintenance
CASE "PAD 4" = PAD()
  lc_new='Y'
  DO MAINTA WITH " 02"

*-- End
CASE "PAD 5" = PAD()
  lc_new='Y'
  DO END WITH " 02"

OTHERWISE
  @ 24,00
  @ 24,21 SAY "This item has no action. Press a key."
  SET CONSOLE OFF
  DO Wait4Key
  SET CONSOLE OFF
  @ 24,00

ENDCASE

SET MESSAGE TO
IF gc_quit='Q'
  IF LEFT(entryflg,1) = "B"
    DEACTIVATE MENU
  ELSE
    DEACTIVATE MENU && MAIN
  ENDIF
ENDIF
IF lc_new='Y'
  lc_file="SET"+gc_prognum
  DO &lc_file.
ENDIF
RETURN
*-- EOP: ACT02 - Menu MAIN

PROCEDURE PROCESS
PARAMETER entryflg
*-----
* Program.....: PROCESS.PRG
* Author.....: K.Sasabe, National River Mouths Study
* Date.....: 8-27-93
* Notice.....: Japan International Cooperation Agency (JICA)
* Generated by.: dBASE IV version 1.5
* Description..: Menu actions
*-----
PRIVATE gc_prognum
gc_prognum="03"

DO SET03
IF gn_error > 0
  gn_error=0
RETURN
ENDIF

ACTIVATE POPUP PROCESS

RETURN
*-- EOP: PROCESS

PROCEDURE SET03
*-----
* PROCESS setup routine. This procedure will open up the file or
* view for this menu, and setup the color scheme.
*-----

ON KEY LABEL F1 DO IHELP1          && Define F1-Help key
DO DBF03                            && open menu level database

IF gn_error = 0
  IF @ color .AND. .NOT. SET("ATTRIBUTES") = ;
    "W/B,RG+/GB,N/N "+CHR(38)+CHR(38)+" RG+/N,W/B,RG+/GB,B/W,N/GB"
    SET COLOR OF NORMAL TO W+/B
    SET COLOR OF MESSAGES TO RG+/N
    SET COLOR OF TITLES TO W/B
    SET COLOR OF HIGHLIGHT TO RG+/GB
    SET COLOR OF BOX TO RG+/GB

```

```

        SET COLOR OF INFORMATION TO B/W
        SET COLOR OF FIELDS TO N/GB
    ENDIF
ENDIF

RETURN
*-- EOP: SET03 - Menu PROCESS

PROCEDURE DBF03
*-----
* PROCESS open DBF or view procedure
*-----
PRIVATE lc message
CLOSE DATABASES
*-- Open menu level view/database
lc message="0"
ON ERROR lc message=LTRIM(STR(ERROR()))+" "+MESSAGE()
USE RESEAR_1.DBF
IF "" <> DBF()
    SET INDEX TO RESEAR_1.MDX
ENDIF
SET ORDER TO SERIAL
ON ERROR
gn_error=VAL(lc_message)
IF gn_error > 0
    DO Pause WITH "Error opening RESEAR_1.DBF or index(es) RESEAR_1.MDX"
    lc_new='Y'
    RETURN
ENDIF
lc_new='Y'
RETURN
*-- EOP: DBF03 - Menu PROCESS

PROCEDURE ACT03
*-----
* Begin PROCESS: POPUP Menu Actions.
* Includes before item, action, and after item as needed
*-----
PRIVATE lc_new, lc_dbf
lc_new=' '
lc_dbf=' '
gc_scope=''
DO CASE

    *-- Amend data
    CASE BAR() = 3
        SAVE WINDOW main TO main.win
        lc_new='Y'
        DO AMEND WITH " 03"
        ACTIVATE WINDOW main
        RESTORE WINDOW main FROM main.win

    *-- Add new records
    CASE BAR() = 4
        SAVE WINDOW main TO main.win
        lc_new='Y'
        DO ADD WITH " 03"
        ACTIVATE WINDOW main
        RESTORE WINDOW main FROM main.win
    ENDCASE

SET MESSAGE TO
IF gc_quit='Q'
    IF LEFT(entryflg,1) = "B"
        DEACTIVATE MENU
    ELSE
        DEACTIVATE POPUP && PROCESS
    ENDIF
ENDIF
IF lc_new='Y'
    lc_file="SET"+gc_prognum
    DO &lc_file.
ENDIF
RETURN
*-- EOP: ACT03 - Menu PROCESS

PROCEDURE CARDVIEW
PARAMETER entryflg
*-----
* Program.....: CARDVIEW.PRG
* Author.....: K.Sasabe, National River Mouths Study
* Date.....: 8-27-93
* Notice.....: Japan International Cooperation Agency (JICA)
* Generated by.: dBASE IV version 1.5
* Description..: Menu actions

```

```

*-----
PRIVATE gc_prognum
gc_prognum="04"

DO SET04
IF gn_error > 0
  gn_error=0
  RETURN
ENDIF

ACTIVATE POPUP CARDVIEW

RETURN
*-- EOP: CARDVIEW

PROCEDURE SET04
*-----
* CARDVIEW setup routine. This procedure will open up the file or
* view for this menu, and setup the color scheme.
*-----

ON KEY LABEL F1 DO 1HELP1          && Define F1-Help key
DO DBF04                          && open menu level database

IF gn_error = 0
  IF gl_color .AND. .NOT. SET("ATTRIBUTES") = ;
    "W/B,RG+/GB,N/N "+CHR(38)+CHR(38)+" RG+/N,W/B,RG+/GB,B/W,N/GB"
    SET COLOR OF NORMAL TO W+/B
    SET COLOR OF MESSAGES TO RG+/N
    SET COLOR OF TITLES TO W/B
    SET COLOR OF HIGHLIGHT TO RG+/GB
    SET COLOR OF BOX TO RG+/GB
    SET COLOR OF INFORMATION TO B/W
    SET COLOR OF FIELDS TO N/GB
  ENDIF
ENDIF

RETURN
*-- EOP: SET04 - Menu CARDVIEW

PROCEDURE DBF04
*-----
* CARDVIEW open DBF or view procedure
*-----
PRIVATE lc_message
CLOSE DATABASES
*-- Open menu level view/database
lc_message="0"
ON ERROR lc_message=LTRIM(STR(ERROR()))+" "+MESSAGE()
USE RESEAR_1.DBF
IF "" <> DBF()
  SET INDEX TO RESEAR_1.MDX
ENDIF
SET ORDER TO SERIAL
ON ERROR
gn_error=VAL(lc_message)
IF gn_error > 0
  DO PAUSE WITH "Error opening RESEAR_1.DBF or index(es) RESEAR_1.MDX"
  lc_new='Y'
  RETURN
ENDIF
lc_new='Y'
RETURN
*-- EOP: DBF04 - Menu CARDVIEW

PROCEDURE ACT04
*-----
* Begin CARDVIEW: POPUP Menu Actions.
* Includes before item, action, and after item as needed
*-----
PRIVATE lc_new, lc_dbf
lc_new=' '
lc_dbf=' '
gc_scope=''
DO CASE

  *-- River mouth name and state
  CASE BAR() = 1
    SAVE WINDOW main TO main.win
    SAVE WINDOW name TO name.win
    SET STATUS OFF
    ACTIVATE WINDOW print
    @ 10,32 SAY "Please wait..."
    IF .NOT. gl_batch

```



```

    DO BefAct
ENDIF
SET MESSAGE TO
*-- Desc: attach format file C_B_INFO
SET FORMAT TO C_B_INFO
EDIT FOR SERIAL=serialno NOFOLLOW NOAPPEND NOMENU NODELETE NOEDIT
*-- close format file so as not to affect READ's
SET FORMAT TO

DEACTIVATE WINDOW print
RESTORE WINDOW name FROM name.win
ACTIVATE WINDOW main
RESTORE WINDOW main FROM main.win
IF .NOT. gl_batch
    DO AftAct
ENDIF

*-- Map/photo data
CASE BAR() = 2
SAVE WINDOW main TO main
SAVE WINDOW name TO name
SET STATUS OFF
ACTIVATE WINDOW print
rno=0
@ 10,32 SAY "Please wait..."
DO OBJRECORD WITH serialno,rno
IF .NOT. gl_batch
    DO BefAct
ENDIF
SET MESSAGE TO
*-- Desc: attach format file MAPPHOT3
SET FORMAT TO MAPPHOT3
EDIT FOR SERIAL=serialno NOFOLLOW NOAPPEND NOMENU NODELETE NOEDIT
*-- close format file so as not to affect READ's
SET FORMAT TO

SELECT 3
ZAP
REINDEX
SELECT 1
DEACTIVATE WINDOW print
RESTORE WINDOW name FROM name.win
ACTIVATE WINDOW main
RESTORE WINDOW main FROM main.win
IF .NOT. gl_batch
    DO AftAct
ENDIF

*-- River features
CASE BAR() = 3
*-- Open Item level view/database and indexes
CLOSE DATABASES
lc dbf='Y'
lc message=""
ON ERROR lc message=LTRIM(STR(ERROR()))+" "+MESSAGE()
SET VIEW TO C_RIVER.QBE
ON ERROR
gn error=VAL(lc message)
IF gn error > 0
    DO Pause WITH "Error opening C_RIVER.QBE"
    gn error=0
    lc file="SET"+gc_prognum
    DO &lc file.
    RETURN
ENDIF
lc new='Y'
SAVE WINDOW main TO main.win
SAVE WINDOW name TO name.win
SET STATUS OFF
ACTIVATE WINDOW print
@ 10,32 SAY "Please wait..."
IF .NOT. gl_batch
    DO BefAct
ENDIF
SET MESSAGE TO
*-- Desc: attach format file C_RIVER
SET FORMAT TO C_RIVER
EDIT FOR SERIAL=serialno NOFOLLOW NOAPPEND NOMENU NODELETE NOEDIT
*-- close format file so as not to affect READ's
SET FORMAT TO

DEACTIVATE WINDOW print
RESTORE WINDOW name FROM name.win

```

```

ACTIVATE WINDOW main
RESTORE WINDOW main FROM main.win
IF .NOT. gl_batch
  DO AftAct
ENDIF

*-- River mouth features...
CASE BAR() = 4
SAVE WINDOW main TO main.win
SAVE WINDOW name TO name.win
lc_new='Y'
DO C MOUTH WITH " 04"
RESTORE WINDOW name FROM name.win
ACTIVATE WINDOW main
RESTORE WINDOW main FROM main.win

*-- Navigation and fishery
CASE BAR() = 5
*-- Open Item level view/database and indexes
CLOSE DATABASES
lc_dbf='Y'
lc_message="0"
ON ERROR lc_message=LTRIM(STR(ERROR()))+" "+MESSAGE()
SET VIEW TO C_NAVI.QBE
ON ERROR
gn_error=VAL(lc_message)
IF gn_error > 0
  DO Pause WITH "Error opening C_NAVI.QBE"
  gn_error=0
  lc_file="SET"+gc_prognum
  DO &lc_file.
  RETURN
ENDIF
lc_new='Y'
SAVE WINDOW main TO main.win
SAVE WINDOW name TO name.win
SET STATUS OFF
ACTIVATE WINDOW print
@ 10,32 SAY "Please wait..."
IF .NOT. gl_batch
  DO BefAct
ENDIF
SET MESSAGE TO
*-- Desc: attach format file C_NAVI
SET FORMAT TO C_NAVI
EDIT FOR SERIAL=serialno NOFOLLOW NOAPPEND NOMENU NODELETE NOEDIT
*-- close format file so as not to affect READ's
SET FORMAT TO

DEACTIVATE WINDOW print
RESTORE WINDOW name FROM name.win
ACTIVATE WINDOW main
RESTORE WINDOW main FROM main.win
IF .NOT. gl_batch
  DO AftAct
ENDIF

*-- Socioeconomic condition...
CASE BAR() = 6
SAVE WINDOW main TO main.win
SAVE WINDOW name TO name.win
lc_new='Y'
DO C SOCIO WITH " 04"
RESTORE WINDOW name FROM name.win
ACTIVATE WINDOW main
RESTORE WINDOW main FROM main.win

*-- Present problems
CASE BAR() = 7
*-- Open Item level view/database and indexes
CLOSE DATABASES
lc_dbf='Y'
lc_message="0"
ON ERROR lc_message=LTRIM(STR(ERROR()))+" "+MESSAGE()
SET VIEW TO C_PROB.QBE
ON ERROR
gn_error=VAL(lc_message)
IF gn_error > 0
  DO Pause WITH "Error opening C_PROB.QBE"
  gn_error=0
  lc_file="SET"+gc_prognum
  DO &lc_file.
  RETURN
ENDIF
lc_new='Y'

```

```

SAVE WINDOW main TO main
SAVE WINDOW name TO name
SET STATUS OFF
ACTIVATE WINDOW print
@ 10,32 SAY "Please wait..."
IF .NOT. gl_batch
  DO BefAct
ENDIF
SET MESSAGE TO
*-- Desc: attach format file C_PROB
SET FORMAT TO C_PROB
EDIT FOR SERIAL=serialno NOFOLLOW NOAPPEND NOMENU NODELETE NOEDIT
*-- close format file so as not to affect READ's
SET FORMAT TO

DEACTIVATE WINDOW print
RESTORE WINDOW name FROM name.win
ACTIVATE WINDOW main
RESTORE WINDOW main FROM main.win
IF .NOT. gl_batch
  DO AftAct
ENDIF

*-- Researcher's comments
CASE BAR() = 8
*-- Open Item level view/database and indexes
CLOSE DATABASES
lc dbf='Y'
lc message="0"
ON ERROR lc message=LTRIM(STR(ERROR()))+" "+MESSAGE()
SET VIEW TO C_COMMEN.QBE
ON ERROR
gn error=VAL(lc message)
IF gn error > 0
  DO Pause WITH "Error opening C_COMMEN.QBE"
  gn error=0
  lc file="SET"+gc_prognum
  DO &lc file.
  RETURN
ENDIF
lc new='Y'
SAVE WINDOW main TO main
SAVE WINDOW name TO name
SET STATUS OFF
ACTIVATE WINDOW print
@ 10,32 SAY "Please wait..."
IF .NOT. gl_batch
  DO BefAct
ENDIF
SET MESSAGE TO
*-- Desc: attach format file C_COMMEN
SET FORMAT TO C_COMMEN
EDIT FOR SERIAL=serialno NOFOLLOW NOAPPEND NOMENU NODELETE NOEDIT
*-- close format file so as not to affect READ's
SET FORMAT TO

DEACTIVATE WINDOW print
RESTORE WINDOW name FROM name.win
ACTIVATE WINDOW main
RESTORE WINDOW main FROM main.win
IF .NOT. gl_batch
  DO AftAct
ENDIF

*-- Categorization and countermeasure
CASE BAR() = 9
*-- Open Item level view/database and indexes
CLOSE DATABASES
lc dbf='Y'
lc message="0"
ON ERROR lc message=LTRIM(STR(ERROR()))+" "+MESSAGE()
SET VIEW TO C_CATEGO.QBE
ON ERROR
gn error=VAL(lc message)
IF gn error > 0
  DO Pause WITH "Error opening C_CATEGO.QBE"
  gn error=0
  lc file="SET"+gc_prognum
  DO &lc file.
  RETURN
ENDIF
lc new='Y'
SAVE WINDOW main TO main
SAVE WINDOW name TO name

```

```

SET STATUS OFF
ACTIVATE WINDOW print
@ 10,32 SAY "Please wait..."
IF .NOT. gl_batch
  DO BefAct
ENDIF
SET MESSAGE TO
*-- Desc: attach format file C_CATEGO
SET FORMAT TO C_CATEGO
EDIT FOR SERIAL=serialno NOFOLLOW NOAPPEND NOMENU NODELETE NOEDIT
*-- close format file so as not to affect READ's
SET FORMAT TO

DEACTIVATE WINDOW print
RESTORE WINDOW name FROM name.win
ACTIVATE WINDOW main
RESTORE WINDOW main FROM main.win
IF .NOT. gl_batch
  DO AftAct
ENDIF
ENDCASE

SET MESSAGE TO
IF gc_quit='Q'
  IF LEFT(entryflg,1) = "B"
    DEACTIVATE MENU
  ELSE
    DEACTIVATE POPUP && CARDVIEW
  ENDIF
ENDIF
IF lc_new='Y'
  lc_file="SET"+gc_prognum
  DO %lc_file.
ENDIF
IF lc_dbf='Y' .AND. .NOT. lc_new='Y'
  lc_file="DBF"+gc_prognum
  DO %lc_file.
ENDIF
RETURN
*-- EOP: ACT04 - Menu CARDVIEW

PROCEDURE BROWVIEW
PARAMETER entryflg
*-----
* Program.....: BROWVIEW.PRG
* Author.....: K.Sasabe, National River Mouths Study
* Date.....: 8-27-93
* Notice.....: Japan International Cooperation Agency (JICA)
* Generated by.: dBASE IV version 1.5
* Description..: Menu actions
*-----
PRIVATE gc_prognum
gc_prognum="05"

DO SET05
IF gn_error > 0
  gn_error=0
RETURN
ENDIF

ACTIVATE POPUP BROWVIEW

RETURN
*-- EOP: BROWVIEW

PROCEDURE SET05
*-----
* BROWVIEW setup routine. This procedure will open up the file or
* view for this menu, and setup the color scheme.
*-----
ON KEY LABEL F1 DO 1HELP1          && Define F1-Help key

DO DBF05                          && open menu level database

IF gn_error = 0
  IF gl_color .AND. .NOT. SET("ATTRIBUTES") = :
    "W+/B, RG+/GB, N/N "+CHR(38)+CHR(38)+ " RG+/N, W/B, RG+/GB, B/W, N/GB"
    SET COLOR OF NORMAL TO W+/B
    SET COLOR OF MESSAGES TO RG+/N
    SET COLOR OF TITLES TO W/B
    SET COLOR OF HIGHLIGHT TO RG+/GB
    SET COLOR OF BOX TO RG+/GB
    SET COLOR OF INFORMATION TO B/W
    SET COLOR OF FIELDS TO N/GB
  
```

```

    ENDIF
  ENDIF

RETURN
*-- EOP: SET05 - Menu BROWVIEW

PROCEDURE DBF05
*-----
* BROWVIEW open DBF or view procedure
*-----
PRIVATE lc_message
CLOSE DATABASES
*-- Open menu level view/database
lc_message="0"
ON_ERROR lc_message=LTRIM(STR(ERROR()))+" "+MESSAGE()
USE RESEAR_1.DBF
IF "" <> DBF()
  SET INDEX TO RESEAR_1.MDX
ENDIF
SET ORDER TO SERIAL
ON_ERROR
gn_error=VAL(lc_message)
IF gn_error > 0
  DO Pause WITH "Error opening RESEAR_1.DBF or index(es) RESEAR_1.MDX"
  lc_new='Y'
RETURN
ENDIF
lc_new='Y'
RETURN
*-- EOP: DBF05 - Menu BROWVIEW

PROCEDURE ACT05
*-----
* Begin BROWVIEW: POPUP Menu Actions.
* Includes before item, action, and after item as needed
*-----
PRIVATE lc_new, lc_dbf
lc_new=' '
lc_dbf=' '
gc_scope=' '
DO CASE

  *-- River mouth list
  CASE BAR() = 1
    *-- Open Item level view/database and indexes
    CLOSE DATABASES
    lc_dbf='Y'
    lc_message="0"
    ON_ERROR lc_message=LTRIM(STR(ERROR()))+" "+MESSAGE()
    SET VIEW TO C_B_INFO.QBE
    ON_ERROR
    gn_error=VAL(lc_message)
    IF gn_error > 0
      DO Pause WITH "Error opening C_B_INFO.QBE"
      gn_error=0
      lc_file="SET"+gc_prognum
      DO &lc_file.
      RETURN
    ENDIF
    lc_new='Y'
    SAVE WINDOW main TO main.win
    SET STATUS OFF
    ACTIVATE WINDOW alarm
    mok="Y"
    @ 0,0 SAY "Printer ready? (Y/N):" GET mok
    READ
    DEACTIVATE WINDOW alarm
    pl=60
    pitch="&160"
    DO P SET WITH mok,pl,pitch    && PROCEDURE NAME should be upper case
    ACTIVATE WINDOW print
    IF .NOT. gl_batch
      DO BefAct
    ENDIF
    SET MESSAGE TO
    *-- Desc: Report
    gn_pkey = 0
    DO PrintSet
    IF gn_pkey <> 27 && esc
      REPORT FORM B_B_INFO NOEJECT &gc_scope.
      DO Cleanup
    ENDIF

DO P_RESET WITH mok

```

```

DEACTIVATE WINDOW print
ACTIVATE WINDOW main
RESTORE WINDOW main FROM main.win
IF .NOT. gl_batch
  DO AftAct
ENDIF

*-- River features
CASE BAR() = 2
*-- Open Item level view/database and indexes
CLOSE DATABASES
lc dbf='Y'
lc message="0"
ON ERROR lc message=LTRIM(STR(ERROR()))+" "+MESSAGE()
SET VIEW TO C_RIVER.QBE
ON ERROR
gn_error=VAL(lc_message)
IF gn_error > 0
  DO Pause WITH "Error opening C_RIVER.QBE"
  gn_error=0
  lc_file="SET"+gc_prognum
  DO &lc_file.
  RETURN
ENDIF
lc new='Y'
SAVE WINDOW main TO main.win
ACTIVATE WINDOW alarm
mok="Y"
@ 0,0 SAY "Printer ready? (Y/N):" GET mok
READ
DEACTIVATE WINDOW alarm
pl=61
pitch="&16D"
DO P SET WITH mok,pl,pitch
ACTIVATE WINDOW print
IF .NOT. gl_batch
  DO BefAct
ENDIF
SET MESSAGE TO
*-- Desc: Report
gn_pkey = 0
DO PrintSet
IF gn_pkey <> 27 && esc
  REPORT FORM B_RIVER NOEJECT &gc_scope.
  DO Cleanup
ENDIF

DO P RESET WITH mok
DEACTIVATE WINDOW print
ACTIVATE WINDOW main
RESTORE WINDOW main FROM main.win
IF .NOT. gl_batch
  DO AftAct
ENDIF

*-- River mouth features...
CASE BAR() = 3
SAVE WINDOW main TO main.win
SET STATUS OFF
lc new='Y'
DO B MOUTH WITH " 05"
ACTIVATE WINDOW main
RESTORE WINDOW main FROM main.win

*-- Navigation and fishery
CASE BAR() = 4
*-- Open Item level view/database and indexes
CLOSE DATABASES
lc dbf='Y'
lc message="0"
ON ERROR lc message=LTRIM(STR(ERROR()))+" "+MESSAGE()
SET VIEW TO C_NAVI.QBE
ON ERROR
gn_error=VAL(lc_message)
IF gn_error > 0
  DO Pause WITH "Error opening C_NAVI.QBE"
  gn_error=0
  lc_file="SET"+gc_prognum
  DO &lc_file.
  RETURN
ENDIF
lc new='Y'
SAVE WINDOW main TO main.win
ACTIVATE WINDOW alarm
mok="Y"

```

```

@ 0,0 SAY "Printer ready? (Y/N):" GET mok
READ
DEACTIVATE WINDOW alarm
pl=67
pitch="&17C"
DO P SET WITH mok,pl,pitch
ACTIVATE WINDOW print
IF .NOT. gl_batch
  DO BefAct
ENDIF
SET MESSAGE TO
*-- Desc: Report
gn_pkey = 0
DO PrintSet
IF gn_pkey <> 27 && esc
  REPORT FORM B_NAVI NOEJECT &gc_scope.
DO Cleanup
ENDIF

DO P RESET WITH mok
DEACTIVATE WINDOW print
ACTIVATE WINDOW main
RESTORE WINDOW main FROM main.win
IF .NOT. gl_batch
  DO AftAct
ENDIF

*-- Socioeconomic conditions...
CASE BAR() = 5
SAVE WINDOW main TO main.win
SET STATUS OFF
lc_new='Y'
DO B SOCIO WITH " 05"
ACTIVATE WINDOW main
RESTORE WINDOW main FROM main.win

*-- Present problems
CASE BAR() = 6
*-- Open Item level view/database and indexes
CLOSE DATABASES
lc_dbf='Y'
lc_message="0"
ON ERROR lc_message=LTRIM(STR(ERROR()))+" "+MESSAGE()
SET VIEW TO C_PROB.QBE
ON ERROR
gn_error=VAL(lc_message)
IF gn_error > 0
  DO Pause WITH "Error opening C_PROB.QBE"
  gn_error=0
  lc_file="SET"+gc_prognum
  DO &lc_file.
  RETURN
ENDIF
lc_new='Y'
SAVE WINDOW main TO main.win
ACTIVATE WINDOW alarm
mok="Y"
@ 0,0 SAY "Printer ready? (Y/N):" GET mok
READ
DEACTIVATE WINDOW alarm
pl=65
pitch="&17.32C"
DO P SET WITH mok,pl,pitch
ACTIVATE WINDOW print
IF .NOT. gl_batch
  DO BefAct
ENDIF
SET MESSAGE TO
*-- Desc: Report
gn_pkey = 0
DO PrintSet
IF gn_pkey <> 27 && esc
  REPORT FORM B_PROB NOEJECT &gc_scope.
DO Cleanup
ENDIF

DO P RESET WITH mok
DEACTIVATE WINDOW print
ACTIVATE WINDOW main
RESTORE WINDOW main FROM main.win
IF .NOT. gl_batch
  DO AftAct
ENDIF

```

```

*-- Categorization and countermeasure
CASE BAR() = 7
*-- Open Item level view/database and indexes
CLOSE DATABASES
lc dbf='Y'
lc message="0"
ON ERROR lc message=LTRIM(STR(ERROR()))+" "+MESSAGE()
SET VIEW TO C_CATEGO.QBE
ON ERROR
gn error=VAL(lc message)
IF gn error > 0
  DO Pause WITH "Error opening C_CATEGO.QBE"
  gn error=0
  lc file="SET"+gc_prognum
  DO &lc_file.
  RETURN
ENDIF
lc new='Y'
SAVE WINDOW main TO main.win
ACTIVATE WINDOW alarm
mok="Y"
@ 0,0 SAY "Printer ready? (Y/N):" GET mok
READ
DEACTIVATE WINDOW alarm
pl=63
pitch="&17.56C"
DO P SET WITH mok,pl,pitch
ACTIVATE WINDOW print
IF .NOT. gl_batch
  DO BefAct
ENDIF
SET MESSAGE TO
*-- Desc: Report
gn pkey = 0
DO PrintSet
IF gn pkey <> 27 && esc
  REPORT FORM B_CATEGO NOEJECT &gc_scope.
  DO Cleanup
ENDIF

DO P RESET WITH mok
DEACTIVATE WINDOW print
ACTIVATE WINDOW main
RESTORE WINDOW main FROM main.win
IF .NOT. gl_batch
  DO AftAct
ENDIF
ENDCASE

SET MESSAGE TO
IF gc quit='Q'
  IF LEFT(entryflg,1) = "B"
    DEACTIVATE MENU
  ELSE
    DEACTIVATE POPUP && BROWVIEW
  ENDIF
ENDIF
IF lc new='Y'
  lc file="SET"+gc_prognum
  DO &lc_file.
ENDIF
IF lc dbf='Y' .AND. .NOT. lc_new='Y'
  lc file="DBF"+gc_prognum
  DO &lc_file.
ENDIF
RETURN
*-- EOP: ACT05 - Menu BROWVIEW

PROCEDURE MAINT
PARAMETER entryflg
*-----
* Program.....: MAINT.PRG
* Author.....: K.Sasabe, National River Mouths Study
* Date.....: 8-27-93
* Notice.....: Japan International Cooperation Agency (JICA)
* Generated by.: dBASE IV version 1.5
* Description...: Menu actions
*-----
PRIVATE gc_prognum
gc_prognum="06"

DO SET06
IF gn error > 0
  gn error=0
  RETURN

```



```

ENDIF
ACTIVATE POPUP MAINTE
RETURN
*-- EOP: MAINTE
PROCEDURE SET06
*-----
* MAINTE setup routine. This procedure will open up the file or
* view for this menu, and setup the color scheme.
*-----
ON KEY LABEL F1 DO IHHELP1          && Define F1-Help key
DO DBF06                            && open menu level database
IF gn_error = 0
  IF g1_color .AND. .NOT. SET("ATTRIBUTES") = ;
    "W/B,RG+/GB,N/N "+CHR(38)+CHR(38)+" RG+/N,W/B,RG+/GB,B/W,N/GB"
    SET COLOR OF NORMAL TO W/B
    SET COLOR OF MESSAGES TO RG+/N
    SET COLOR OF TITLES TO W/B
    SET COLOR OF HIGHLIGHT TO RG+/GB
    SET COLOR OF BOX TO RG+/GB
    SET COLOR OF INFORMATION TO B/W
    SET COLOR OF FIELDS TO N/GB
  ENDIF
ENDIF
RETURN
*-- EOP: SET06 - Menu MAINTE
PROCEDURE DBF06
*-----
* MAINTE open DBF or view procedure
*-----
PRIVATE lc_message
CLOSE DATABASES
*-- Open menu level view/database
lc_message=""
ON ERROR lc_message=LTRIM(STR(ERROR()))+" "+MESSAGE()
USE RESEAR_1.DBF
IF "" <> DBF()
  SET INDEX TO RESEAR_1.MDX
ENDIF
SET ORDER TO SERIAL
ON ERROR
gn_error=VAL(lc_message)
IF gn_error > 0
  DO Pause WITH "Error opening RESEAR_1.DBF or index(es) RESEAR_1.MDX"
  lc_new='Y'
  RETURN
ENDIF
lc_new='Y'
RETURN
*-- EOP: DBF06 - Menu MAINTE
PROCEDURE ACT06
*-----
* Begin MAINTE: POPUP Menu Actions.
* Includes before item, action, and after item as needed
*-----
PRIVATE lc_new, lc_dbf
lc_new='Y'
lc_dbf=' '
gc_scope=''
DO CASE
  *-- Change password
  CASE BAR() = 1
    SAVE WINDOW main TO main2
    SET STATUS OFF
    IF .NOT. g1_batch
      DO BefAct
    ENDIF
    SET MESSAGE TO
    *-- Desc: Inline DO dBASE commands
    DO PASSNEW
    ACTIVATE WINDOW main
    RESTORE WINDOW main FROM main2
    IF .NOT. g1_batch
      DO AftAct
    ENDIF
  OTHERWISE
    *
  ENDIF

```

```

*-- Back up file to floppy
CASE BAR() = 2
  SAVE WINDOW main TO main
  mok="N"
  ACTIVATE WINDOW alarm
  @ 0,0 SAY "Are you sure to backup files? (Y/N):" GET mok
  READ
  DEACTIVATE WINDOW alarm
  IF mok="N"
    RETURN
  ENDIF
  ACTIVATE WINDOW print
  IF .NOT. gl_batch
    DO BefAct
  ENDIF
  SET MESSAGE TO
  RUN BACKUP E:\COPY\*. * A: /S

  DEACTIVATE WINDOW print
  ACTIVATE WINDOW main
  RESTORE WINDOW main FROM main
  IF .NOT. gl_batch
    DO AftAct
  ENDIF

*-- Restore file from floppy
CASE BAR() = 3:
  SAVE WINDOW main TO main
  mok="N"
  ACTIVATE WINDOW alarm
  @ 0,0 SAY "Are you sure to restore files? (Y/N):" GET mok
  READ
  DEACTIVATE WINDOW alarm
  IF mok="N"
    RETURN
  ENDIF
  ACTIVATE WINDOW print
  IF .NOT. gl_batch
    DO BefAct
  ENDIF
  SET MESSAGE TO
  RUN RESTORE A: E:\COPY\*. * /S

  DEACTIVATE WINDOW print
  ACTIVATE WINDOW main
  RESTORE WINDOW main FROM main
  IF .NOT. gl_batch
    DO AftAct
  ENDIF
ENDCASE

SET MESSAGE TO
IF gc_quit='Q'
  IF LEFT(entryflg,1) = "B"
    DEACTIVATE MENU
  ELSE
    DEACTIVATE POPUP.&& MAINTEN
  ENDIF
ENDIF
IF lc_new='Y'
  lc_file="SET"+gc_prognum
  DO &lc_file.
ENDIF
RETURN
*-- EOP: ACT06 - Menu MAINTEN

PROCEDURE END
PARAMETER entryflg
*-----
* Program.....: END.PRG
* Author.....: K.Sasabe, National River Mouths Study
* Date.....: 8-27-93
* Notice.....: Japan International Cooperation Agency (JICA)
* Generated by.: dBASE IV version 1.5
* Description..: Menu actions
*-----
PRIVATE gc_prognum
gc_prognum="07"

DO SET07
IF gn_error > 0
  gn_error=0
RETURN
ENDIF

ACTIVATE POPUP END

```

```

RETURN
*-- EOP: END

PROCEDURE SET07
*-----
* END setup routine. This procedure will open up the file or
* view for this menu, and setup the color scheme.
*-----
ON KEY LABEL F1 DO 1HELP1           && Define F1-Help key
DO DBF07                             && open menu level database

IF gn_error = 0
  IF gl_color .AND. .NOT. SET("ATTRIBUTES") = ;
    "W+/B,RG+/GB,N/N "+CHR(38)+CHR(38)+" RG+/N,W/B,RG+/GB,B/W,N/GB"
    SET COLOR OF NORMAL TO W+/B
    SET COLOR OF MESSAGES TO RG+/N
    SET COLOR OF TITLES TO W/B
    SET COLOR OF HIGHLIGHT TO RG+/GB
    SET COLOR OF BOX TO RG+/GB
    SET COLOR OF INFORMATION TO B/W
    SET COLOR OF FIELDS TO N/GB
  ENDIF
ENDIF

RETURN
*-- EOP: SET07 - Menu END

PROCEDURE DBF07
*-----
* END open DBF or view procedure
*-----
PRIVATE lc_message
CLOSE DATABASES
*-- Open menu level view/database
lc_message="0"
ON ERROR lc_message=LTRIM(STR(ERROR()))+" "+MESSAGE()
USE RESEAR_1.DBF
IF "" <> DBF()
  SET INDEX TO RESEAR_1.MDX
ENDIF
SET ORDER TO SERIAL
ON ERROR
gn_error=VAL(lc_message)
IF gn_error > 0
  DO PAUSE WITH "Error opening RESEAR_1.DBF or index(es) RESEAR_1.MDX"
  lc_new='Y'
  RETURN
ENDIF
lc_new='Y'
RETURN
*-- EOP: DBF07 - Menu END

PROCEDURE ACT07
*-----
* Begin END: POPUP Menu Actions.
* Includes before item, action, and after item as needed
*-----
PRIVATE lc_new, lc_dbf
lc_new=' '
lc_dbf=' '
gc_scope=''
DO CASE

  *-- End session
  CASE BAR() = 3
    *-- Return to caller
    gc_quit='Q'
    IF LEFT(entryflg,1) <> "B"
      DEACTIVATE POPUP && END
    ELSE
      DEACTIVATE MENU
    ENDIF
  RETURN

  *-- Quit to DOS
  CASE BAR() = 4
    *-- Quit dBASE
    CLOSE DATABASES
    QUIT
  ENDCASE

SET MESSAGE TO

```

```

IF gc quit='Q'
  IF LEFT(entryflg,1) = "B"
    DEACTIVATE MENU
  ELSE
    DEACTIVATE POPUP && END
  ENDIF
ENDIF
IF lc new='Y'
  lc file="SET"+gc_prognum
  DO &lc_file.
ENDIF
RETURN
*-- EOP: ACT07 - Menu END

```

```

PROCEDURE AMEND
PARAMETER entryflg

```

```

*-----
* Program.....: AMEND.PRG
* Author.....: K.Sasabe, National River Mouths Study
* Date.....: 8-27-93
* Notice.....: Japan International Cooperation Agency (JICA)
* Generated by.: dBASE IV version 1.5
* Description..: Menu actions
*-----

```

```

PRIVATE gc_prognum
gc_prognum="08"

```

```

DO SET08
IF gn_error > 0
  gn_error=0
RETURN
ENDIF

```

```

ACTIVATE POPUP AMEND

```

```

RETURN
*-- EOP: AMEND

```

```

PROCEDURE SET08

```

```

*-----
* AMEND setup routine. This procedure will open up the file or
* view for this menu, and setup the color scheme.
*-----

```

```

ON KEY LABEL F1 DO 1HELP1          && Define F1-Help key
DO DBF08                          && open menu level database

```

```

IF gn_error = 0
  IF .NOT. color .AND. .NOT. SET("ATTRIBUTES") = ;
    "W/B,RG+/GB,N/N "+CHR(38)+CHR(38)+" RG+/N,W/B,RG+/GB,B/W,N/GB"
    SET COLOR OF NORMAL TO W+/B
    SET COLOR OF MESSAGES TO RG+/N
    SET COLOR OF TITLES TO W/B
    SET COLOR OF HIGHLIGHT TO RG+/GB
    SET COLOR OF BOX TO RG+/GB
    SET COLOR OF INFORMATION TO B/W
    SET COLOR OF FIELDS TO N/GB
  ENDIF
ENDIF

```

```

RETURN
*-- EOP: SET08 - Menu AMEND

```

```

PROCEDURE DBF08

```

```

*-----
* AMEND open DBF or view procedure
*-----

```

```

PRIVATE lc_message
CLOSE DATABASES
*-- Open menu level view/database
lc_message="0"
ON ERROR lc_message=LTRIM(STR(ERROR()))+" "+MESSAGE()
USE RESEAR_1.DBF
IF "" <> DBF()
  SET INDEX TO RESEAR_1.MDX
ENDIF
SET ORDER TO SERIAL
ON ERROR
gn_error=VAL(lc_message)
IF gn_error > 0
  DO Pause WITH "Error opening RESEAR_1.DBF or index(es) RESEAR_1.MDX"
  lc_new='Y'
RETURN
ENDIF
lc_new='Y'

```

```

RETURN
*-- EOP: DBF08 - Menu AMEND

PROCEDURE ACT08
*-----
* Begin AMEND: POPUP Menu Actions.
* Includes before item, action, and after item as needed
*-----
PRIVATE lc_new, lc_dbf
lc_new=' '
lc_dbf=' '
gc_scope=' '
DO CASE

*-- RESEAR_1
CASE BAR() = 1
  SAVE WINDOW main TO main2
  lc_new='Y'
  DO RESEAR_1 WITH " 08"
  ACTIVATE WINDOW main
  RESTORE WINDOW main FROM main2.WIN

*-- RES_1_F1
CASE BAR() = 2
  *-- Open Item level view/database and indexes
  CLOSE DATABASES
  lc_dbf='Y'
  lc_message="0"
  ON ERROR lc_message=LTRIM(STR(ERROR()))+" "+MESSAGE()
  SET VIEW TO RES_1_F1.QBE
  ON ERROR
  gn_error=VAL(lc_message)
  IF gn_error > 0
    DO Pause WITH "Error opening RES_1_F1.QBE"
    gn_error=0
    lc_file="SET"+gc_prognum
    DO &lc_file.
    RETURN
  ENDIF
  lc_new='Y'
  SAVE WINDOW main TO main2
  SET STATUS ON
  ACTIVATE WINDOW print
  IF .NOT. gl_batch
    DO BefAct
  ENDIF
  SET MESSAGE TO
  *-- Desc: attach format file RES_1_F1
  SET FORMAT TO RES_1_F1
  EDIT &gc_scope. NOFOLLOW NOAPPEND NODELETE
  *-- close format file so as not to affect READ's
  SET FORMAT TO

  DEACTIVATE WINDOW print
  SET STATUS OFF
  ACTIVATE WINDOW main
  RESTORE WINDOW main FROM main2
  IF .NOT. gl_batch
    DO AftAct
  ENDIF

*-- RES_1_F2
CASE BAR() = 3
  *-- Open Item level view/database and indexes
  CLOSE DATABASES
  lc_dbf='Y'
  lc_message="0"
  ON ERROR lc_message=LTRIM(STR(ERROR()))+" "+MESSAGE()
  SET VIEW TO RES_1_F2.QBE
  ON ERROR
  gn_error=VAL(lc_message)
  IF gn_error > 0
    DO Pause WITH "Error opening RES_1_F2.QBE"
    gn_error=0
    lc_file="SET"+gc_prognum
    DO &lc_file.
    RETURN
  ENDIF
  lc_new='Y'
  SAVE WINDOW main TO main2
  SET STATUS ON
  ACTIVATE WINDOW print
  IF .NOT. gl_batch
    DO BefAct

```

```

ENDIF
SET MESSAGE TO
*-- Desc: attach format file RES_1_F2
SET FORMAT TO RES_1_F2
EDIT &gc scope. NOFOLLOW NOAPPEND NODELETE
*-- close format file so as not to affect READ's
SET FORMAT TO

DEACTIVATE WINDOW print
SET STATUS OFF
ACTIVATE WINDOW main
RESTORE WINDOW main FROM main2
IF .NOT. gl_batch
DO AftAct
ENDIF

*-- RES 1 F3
CASE BAR()= 4
*-- Open Item level view/database and indexes
CLOSE DATABASES
lc dbf='Y'
lc_message="0"
ON ERROR lc_message=LTRIM(STR(ERROR()))+" "+MESSAGE()
SET VIEW TO RES_1_F3.QBE
ON ERROR
gn_error=VAL(lc_message)
IF gn_error > 0
DO Pause WITH "Error opening RES_1_F3.QBE"
gn_error=0
lc file="SET"+gc_prognum
DO &lc file.
RETURN
ENDIF
lc new='Y'
SAVE WINDOW main TO main2
SET STATUS ON
ACTIVATE WINDOW print
IF .NOT. gl_batch
DO BefAct
ENDIF
SET MESSAGE TO
*-- Desc: attach format file RES_1_F3
SET FORMAT TO RES_1_F3
EDIT &gc scope. NOFOLLOW NOAPPEND NODELETE
*-- close format file so as not to affect READ's
SET FORMAT TO

DEACTIVATE WINDOW print
SET STATUS OFF
ACTIVATE WINDOW main
RESTORE WINDOW main FROM main2
IF .NOT. gl_batch
DO AftAct
ENDIF

*-- RES 1 F4
CASE BAR()= 5
*-- Open Item level view/database and indexes
CLOSE DATABASES
lc dbf='Y'
lc_message="0"
ON ERROR lc_message=LTRIM(STR(ERROR()))+" "+MESSAGE()
SET VIEW TO RES_1_F4.QBE
ON ERROR
gn_error=VAL(lc_message)
IF gn_error > 0
DO Pause WITH "Error opening RES_1_F4.QBE"
gn_error=0
lc file="SET"+gc_prognum
DO &lc file.
RETURN
ENDIF
lc new='Y'
SAVE WINDOW main TO main2
SET STATUS ON
ACTIVATE WINDOW print
IF .NOT. gl_batch
DO BefAct
ENDIF
SET MESSAGE TO
*-- Desc: attach format file RES_1_F4
SET FORMAT TO RES_1_F4
EDIT &gc scope. NOFOLLOW NOAPPEND NODELETE
*-- close format file so as not to affect READ's

```

```
SET FORMAT TO
```

```
DEACTIVATE WINDOW print  
SET STATUS OFF  
ACTIVATE WINDOW main  
RESTORE WINDOW main FROM main2  
IF .NOT. gl_batch  
  DO AftAct  
ENDIF
```

```
*-- RES 1 F5  
CASE BART() = 6  
*-- Open Item level view/database and indexes  
CLOSE DATABASES  
lc_dbf='Y'  
lc_message="0"  
ON ERROR lc_message=LTRIM(STR(ERROR()))+" "+MESSAGE()  
SET VIEW TO RES_1_F5.QBE  
ON ERROR  
gn_error=VAL(lc_message)  
IF gn_error > 0  
  DO Pause WITH "Error opening RES_1_F5.QBE"  
  gn_error=0  
  lc_file="SET"+gc_prognum  
  DO &lc_file.  
  RETURN  
ENDIF  
lc_new='Y'  
SAVE WINDOW main TO main2  
SET STATUS OFF  
ACTIVATE WINDOW print  
IF .NOT. gl_batch  
  DO BefAct  
ENDIF  
SET MESSAGE TO  
*-- Desc: attach format file RES_1_F5  
SET FORMAT TO RES 1 F5  
EDIT &gc_scope. NOFOLLOW NOAPPEND NODELETE  
*-- close format file so as not to affect READ's  
SET FORMAT TO
```

```
DEACTIVATE WINDOW print  
SET STATUS OFF  
ACTIVATE WINDOW main  
RESTORE WINDOW main FROM main2  
IF .NOT. gl_batch  
  DO AftAct  
ENDIF
```

```
*-- RES 1 F6  
CASE BART() = 7  
*-- Open Item level view/database and indexes  
CLOSE DATABASES  
lc_dbf='Y'  
lc_message="0"  
ON ERROR lc_message=LTRIM(STR(ERROR()))+" "+MESSAGE()  
SET VIEW TO RES_1_F6.QBE  
ON ERROR  
gn_error=VAL(lc_message)  
IF gn_error > 0  
  DO Pause WITH "Error opening RES_1_F6.QBE"  
  gn_error=0  
  lc_file="SET"+gc_prognum  
  DO &lc_file.  
  RETURN  
ENDIF  
lc_new='Y'  
SAVE WINDOW main TO main2  
SET STATUS ON  
ACTIVATE WINDOW print  
IF .NOT. gl_batch  
  DO BefAct  
ENDIF  
SET MESSAGE TO  
*-- Desc: attach format file RES_1_F6  
SET FORMAT TO RES 1 F6  
EDIT &gc_scope. NOFOLLOW NOAPPEND NODELETE  
*-- close format file so as not to affect READ's  
SET FORMAT TO
```

```
DEACTIVATE WINDOW print  
SET STATUS OFF  
ACTIVATE WINDOW main
```

```

RESTORE WINDOW main FROM main2
IF .NOT. gl_batch
  DO AftAct
ENDIF

*-- RESEAR 2
CASE BAR() = 8
SAVE WINDOW main TO main2
lc new='Y'
DO RESEAR 2 WITH " 08"
ACTIVATE WINDOW main
RESTORE WINDOW main FROM main2

*-- RESEAR 3
CASE BAR() = 9
SAVE WINDOW main TO main2
lc new='Y'
DO RESEAR 3 WITH " 08"
ACTIVATE WINDOW main
RESTORE WINDOW main FROM main2

*-- ADD INFO
CASE BAR() = 10
*-- Open Item level view/database and indexes
CLOSE DATABASES
lc dbf='Y'
lc message="0"
ON ERROR lc message=LTRIM(STR(ERROR()))+" "+MESSAGE()
SET VIEW TO ADD_INFO.QBE
ON ERROR
gn error=VAL(lc message)
IF gn error > 0
  DO Pause WITH "Error opening ADD_INFO.QBE"
  gn error=0
  lc file="SET"+gc_prognum
  DO &lc file.
  RETURN
ENDIF
lc new='Y'
SAVE WINDOW main TO main2
SET STATUS ON
ACTIVATE WINDOW print
IF .NOT. gl_batch
  DO BefAct
ENDIF
SET MESSAGE TO
*-- Desc: attach format file ADD_INFO
SET FORMAT TO ADD_INFO
EDIT &gc scope. NOFOLLOW NOAPPEND NODELETE
*-- close format file so as not to affect READ's
SET FORMAT TO

DEACTIVATE WINDOW print
SET STATUS OFF
ACTIVATE WINDOW main
RESTORE WINDOW main FROM main2
IF .NOT. gl_batch
  DO AftAct
ENDIF

*-- MAPPHOTO
CASE BAR() = 11
*-- Open Item level view/database and indexes
CLOSE DATABASES
lc dbf='Y'
lc message="0"
ON ERROR lc message=LTRIM(STR(ERROR()))+" "+MESSAGE()
SET VIEW TO MAPPHOTO.QBE
ON ERROR
gn error=VAL(lc message)
IF gn error > 0
  DO Pause WITH "Error opening MAPPHOTO.QBE"
  gn error=0
  lc file="SET"+gc_prognum
  DO &lc file.
  RETURN
ENDIF
lc new='Y'
SAVE WINDOW main TO main2
SET STATUS ON
ACTIVATE WINDOW print
IF .NOT. gl_batch
  DO BefAct
ENDIF
SET MESSAGE TO

```



```

*-- Desc: attach format file MAPPHOTO
SET FORMAT TO MAPPHOTO
EDIT &gc scope. NOFOLLOW NOAPPEND NODELETE
*-- close format file so as not to affect READ's
SET FORMAT TO

DEACTIVATE WINDOW print
SET STATUS OFF
ACTIVATE WINDOW main
RESTORE WINDOW main FROM main2
IF .NOT. gl_batch
DO AftAct
ENDIF
ENDCASE

SET MESSAGE TO
IF gc_quit='Q'
IF LEFT(entryflg,1) = "B"
DEACTIVATE MENU
ELSE
DEACTIVATE POPUP && AMEND
ENDIF
ENDIF
IF lc_new='Y'
lc_file="SET"+gc_prognum
DO &lc_file.
ENDIF
IF lc_dbf='Y' .AND. .NOT. lc_new='Y'
lc_file="DBF"+gc_prognum
DO &lc_file.
ENDIF
RETURN
*-- EOP: ACT08 - Menu AMEND

PROCEDURE ADD
PARAMETER entryflg
-----
* Program.....: ADD.PRG
* Author.....: K.Sasabe, National River Mouths Study
* Date.....: 8-27-93
* Notice.....: Japan International Cooperation Agency (JICA)
* Generated by.: dBASE IV version 1.5
* Description..: Menu actions
-----
PRIVATE gc_prognum
gc_prognum="09"

DO SET09
IF gn_error > 0
gn_error=0
RETURN
ENDIF

ACTIVATE POPUP ADD

RETURN
*-- EOP: ADD

PROCEDURE SET09
-----
* ADD setup routine. This procedure will open up the file or
* view for this menu, and setup the color scheme.
-----

ON KEY LABEL F1 DO 1HELP1          && Define F1-Help key

DO DBF09                            && open menu level database

IF gn_error = 0
IF gl_color .AND. .NOT. SET("ATTRIBUTES") = ;
"W+/B,RG+/GB,N/N "+CHR(38)+CHR(38)+" RG+/N,W/B,RG+/GB,B/W,N/GB"
SET COLOR OF NORMAL TO W+/B
SET COLOR OF MESSAGES TO RG+/N
SET COLOR OF TITLES TO W/B
SET COLOR OF HIGHLIGHT TO RG+/GB
SET COLOR OF BOX TO RG+/GB
SET COLOR OF INFORMATION TO B/W
SET COLOR OF FIELDS TO N/GB
ENDIF
ENDIF

RETURN
*-- EOP: SET09 - Menu ADD

PROCEDURE DBF09

```

```

*-----
* ADD open DBF or view procedure
*-----
PRIVATE lc_message
CLOSE DATABASES
*-- Open menu level view/database
lc_message="0"
ON ERROR lc_message=LTRIM(STR(ERROR()))+" "+MESSAGE()
USE RESEAR_1.DBF
IF "" <> DBF()
  SET INDEX TO RESEAR_1.MDX
ENDIF
SET ORDER TO SERIAL
ON ERROR
gn_error=VAL(lc_message)
IF gn_error > 0
  DO Pause WITH "Error opening RESEAR_1.DBF or index(es) RESEAR_1.MDX"
  lc_new='Y'
  RETURN
ENDIF
lc_new='Y'
RETURN
*-- EOP: DBF09 - Menu ADD

PROCEDURE ACT09
*-----
* Begin ADD: POPUP Menu Actions.
* Includes before item, action, and after item as needed
*-----
PRIVATE lc_new, lc_dbf
lc_new='Y'
lc_dbf=' '
gc_scope=''
DO CASE

*-- MAPPHOTO
CASE BAR() = 1
  *-- Open Item level view/database and indexes
  CLOSE DATABASES
  lc_dbf='Y'
  lc_message="0"
  ON ERROR lc_message=LTRIM(STR(ERROR()))+" "+MESSAGE()
  USE MAPPHOTO.DBF
  ON ERROR
  gn_error=VAL(lc_message)
  IF gn_error > 0
    DO Pause WITH "Error opening MAPPHOTO.DBF"
    gn_error=0
    lc_file="SET"+gc_prognum
    DO &lc_file.
    RETURN
  ENDIF
  lc_new='Y'
  SAVE WINDOW main TO main2
  ACTIVATE WINDOW print
  SET STATUS ON
  IF .NOT. gl_batch
    DO BefAct
  ENDIF
  SET MESSAGE TO
  *-- Desc: attach format file MAPPHOT2
  SET FORMAT TO MAPPHOT2
  APPEND

  *-- close format file so as not to affect READ's
  SET FORMAT TO

  DEACTIVATE WINDOW print
  SET STATUS OFF
  ACTIVATE WINDOW main
  RESTORE WINDOW main FROM main2
  IF .NOT. gl_batch
    DO AftAct
  ENDIF
ENDCASE

SET MESSAGE TO
IF gc_quit='Q'
  IF LEFT(entryflg,1) = "B"
    DEACTIVATE MENU
  ELSE
    DEACTIVATE POPUP && ADD
  ENDIF
ENDIF

```

```

IF lc_new='Y'
  lc_file="SET"+gc_prognum
  DO &lc_file.
ENDIF
IF lc_dbf='Y' .AND. .NOT. lc_new='Y'
  lc_file="DBF"+gc_prognum
  DO &lc_file.
ENDIF
RETURN
*-- EOP: ACT09 - Menu ADD

PROCEDURE C_MOUTH
PARAMETER entryflg
*-----
* Program.....: C_MOUTH.PRG
* Author.....: K.Sasabe, National River Mouths Study
* Date.....: 8-27-93
* Notice.....: Japan International Cooperation Agency (JICA)
* Generated by.: dBASE IV version 1.5
* Description..: Menu actions
*-----
PRIVATE gc_prognum
gc_prognum="010"

DO SET010
IF gn_error > 0
  gn_error=0
  RETURN
ENDIF

ACTIVATE POPUP C_MOUTH

RETURN
*-- EOP: C_MOUTH

PROCEDURE SET010
*-----
* C_MOUTH setup routine. This procedure will open up the file or
* view for this menu, and setup the color scheme.
*-----

ON KEY LABEL F1 DO IHELP1          && Define F1-Help key

DO DBF010                          && open menu level database

IF gn_error = 0
  IF .NOT. color .AND. .NOT. SET("ATTRIBUTES") = ;
    "W+/B,RG+/GB,N/N "+CHR(38)+CHR(38)+" RG+/N,W/B,RG+/GB,B/W,N/GB"
    SET COLOR OF NORMAL TO W+/B
    SET COLOR OF MESSAGES TO RG+/N
    SET COLOR OF TITLES TO W/B
    SET COLOR OF HIGHLIGHT TO RG+/GB
    SET COLOR OF BOX TO RG+/GB
    SET COLOR OF INFORMATION TO B/W
    SET COLOR OF FIELDS TO N/GB
  ENDIF
ENDIF

RETURN
*-- EOP: SET010 - Menu C_MOUTH

PROCEDURE DBF010
*-----
* C_MOUTH open DBF or view procedure
*-----
PRIVATE lc_message
CLOSE DATABASES
*-- Open menu level view/database
lc_message="0"
ON ERROR lc_message=LTRIM(STR(ERROR()))+" "+MESSAGE()
USE RESEAR_1.DBF
IF "" <> DBF()
  SET INDEX TO RESEAR_1.MDX
ENDIF
SET ORDER TO SERIAL
ON ERROR
gn_error=VAL(lc_message)
IF gn_error > 0
  DO Pause WITH "Error opening RESEAR_1.DBF or index(es) RESEAR_1.MDX"
  lc_new='Y'
  RETURN
ENDIF
lc_new='Y'
RETURN
*-- EOP: DBF010 - Menu C_MOUTH

```

```

PROCEDURE ACT010
*-----
* Begin C MOUTH: POPUP Menu Actions.
* Includes before item, action, and after item as needed
*-----
PRIVATE lc_new, lc_dbf
lc_new=' '
lc_dbf=' '
gc_scope=''
DO CASE

  *-- Geomorphology
CASE BAR() = 1
  *-- Open Item level view/database and indexes
  CLOSE DATABASES
  lc_dbf='Y'
  lc_message="0"
  ON ERROR lc_message=LTRIM(STR(ERROR()))+" "+MESSAGE()
  SET VIEW TO C_M_GEO.QBE
  ON ERROR
  gn_error=VAL(lc_message)
  IF gn_error > 0
    DO Pause WITH "Error opening C_M_GEO.QBE"
    gn_error=0
    lc_file="SET"+gc_prognum
    DO &lc_file.
    RETURN
  ENDIF
  lc_new='Y'
  SAVE WINDOW main TO main2
  SET STATUS OFF
  ACTIVATE WINDOW print
  @ 10,32 SAY "Please wait..."
  IF .NOT. gl_batch
    DO BefAct
  ENDIF
  SET MESSAGE TO
  *-- Desc: attach format file C_M_GEO
  SET FORMAT TO C_M_GEO
  EDIT FOR SERIAL=serialno NOFOLLOW NOAPPEND NOMENU NODELETE NOEDIT
  *-- close format file so as not to affect READ's
  SET FORMAT TO

  DEACTIVATE WINDOW print
  RESTORE WINDOW name FROM name.win
  ACTIVATE WINDOW main
  RESTORE WINDOW main FROM main2.win
  IF .NOT. gl_batch
    DO AftAct
  ENDIF

  *-- Oceanographic data
CASE BAR() = 2
  *-- Open Item level view/database and indexes
  CLOSE DATABASES
  lc_dbf='Y'
  lc_message="0"
  ON ERROR lc_message=LTRIM(STR(ERROR()))+" "+MESSAGE()
  SET VIEW TO C_M_OCEA.QBE
  ON ERROR
  gn_error=VAL(lc_message)
  IF gn_error > 0
    DO Pause WITH "Error opening C_M_OCEA.QBE"
    gn_error=0
    lc_file="SET"+gc_prognum
    DO &lc_file.
    RETURN
  ENDIF
  lc_new='Y'
  SAVE WINDOW main TO main2
  SET STATUS OFF
  ACTIVATE WINDOW print
  @ 10,32 SAY "Please wait..."
  IF .NOT. gl_batch
    DO BefAct
  ENDIF
  SET MESSAGE TO
  *-- Desc: attach format file C_M_OCEA
  SET FORMAT TO C_M_OCEA
  EDIT FOR SERIAL=serialno NOFOLLOW NOAPPEND NOMENU NODELETE NOEDIT
  *-- close format file so as not to affect READ's
  SET FORMAT TO

```

```

DEACTIVATE WINDOW print
RESTORE WINDOW name FROM name.win
ACTIVATE WINDOW main
RESTORE WINDOW main FROM main2.win
IF .NOT. gl_batch
  DO AftAct
ENDIF

*-- Structures...
CASE BAR() = 3
  SAVE WINDOW main TO main2
  SET STATUS OFF
  lc_new='Y'
  DO C M STR WITH " 010"
  RESTORE WINDOW name FROM name.win
  ACTIVATE WINDOW main
  RESTORE WINDOW main FROM main2
ENDCASE

SET MESSAGE TO
IF gc_quit='Q'
  IF LEFT(entryflg,1) = "B"
    DEACTIVATE MENU
  ELSE
    DEACTIVATE POPUP && C_MOUTH
  ENDIF
ENDIF
IF lc_new='Y'
  lc_file="SET"+gc_prognum
  DO &lc_file.
ENDIF
IF lc_dbf='Y' .AND. .NOT. lc_new='Y'
  lc_file="DBF"+gc_prognum
  DO &lc_file.
ENDIF
RETURN
*-- EOP: ACT010 - Menu C_MOUTH

PROCEDURE C_SOCIO
PARAMETER eentryflg
*-----
* Program.....: C_SOCIO.PRG
* Author.....: K.Sasabe, National River Mouths Study
* Date.....: 8-27-93
* Notice.....: Japan International Cooperation Agency (JICA)
* Generated by.: dBASE IV version 1.5
* Description..: Menu actions
*-----
PRIVATE gc_prognum
gc_prognum="011"

DO SET011
IF gn_error > 0
  gn_error=0
  RETURN
ENDIF

ACTIVATE POPUP C_SOCIO

RETURN
*-- EOP: C_SOCIO

PROCEDURE SET011
*-----
* C_SOCIO setup routine. This procedure will open up the file or
* view for this menu, and setup the color scheme.
*-----
ON KEY LABEL F1 DO IHELP1          && Define F1-Help key

DO DBF011                          && open menu level database

IF gn_error = 0
  IF gl_color .AND. .NOT. SET("ATTRIBUTES") = ;
    "W+/B,RG+/GB,N/N "+CHR(38)+CHR(38)+" RG+/N,W/B,RG+/GB,B/W,N/GB"
    SET COLOR OF NORMAL TO W+/B
    SET COLOR OF MESSAGES TO RG+/N
    SET COLOR OF TITLES TO W/B
    SET COLOR OF HIGHLIGHT TO RG+/GB
    SET COLOR OF BOX TO RG+/GB
    SET COLOR OF INFORMATION TO B/W
    SET COLOR OF FIELDS TO N/GB
  ENDIF
ENDIF

RETURN

```

```

*-- EOP: SET011 - Menu C_SOCIO
PROCEDURE DBF011
*-----
* C_SOCIO open DBF or view procedure
*-----
PRIVATE lc_message
CLOSE DATABASES
*-- Open menu level view/database
lc_message="0"
ON ERROR lc_message=LTRIM(STR(ERROR()))+" "+MESSAGE()
USE RESEAR_1.DBF
IF "" <> DBF()
    SET INDEX TO RESEAR_1.MDX
ENDIF
SET ORDER TO SERIAL
ON ERROR
gn_error=VAL(lc_message)
IF gn_error > 0
    DO Pause WITH "Error opening RESEAR_1.DBF or index(es) RESEAR_1.MDX"
    lc_new='Y'
RETURN
ENDIF
lc_new='Y'
RETURN
*-- EOP: DBF011 - Menu C_SOCIO

PROCEDURE ACT011
*-----
* Begin C_SOCIO: POPUP Menu Actions.
* IncludeS before item, action, and after item as needed
*-----
PRIVATE lc_new, lc_dbf
lc_new=' '
lc_dbf=' '
gc_scope=' '
DO CASE

*-- Water use
CASE BAR() = 1
*-- Open Item level view/database and indexes
CLOSE DATABASES
lc_dbf='Y'
lc_message="0"
ON ERROR lc_message=LTRIM(STR(ERROR()))+" "+MESSAGE()
SET VIEW TO C_S_WUSE.QBE
ON ERROR
gn_error=VAL(lc_message)
IF gn_error > 0
    DO Pause WITH "Error opening C_S_WUSE.QBE"
    gn_error=0
    lc_file="SET"+gc_prognum
    DO &lc_file.
    RETURN
ENDIF
lc_new='Y'
SAVE WINDOW main TO main2
SET STATUS OFF
ACTIVATE WINDOW print
@ 10,32 SAY "Please wait..."
IF .NOT. gl_batch
    DO BefAct
ENDIF
SET MESSAGE TO
*-- Desc: attach format file C_S_WUSE
SET FORMAT TO C_S_WUSE
EDIT FOR SERIAL=serialno NOFOLLOW NOAPPEND NOMENU NODELETE NOEDIT
*-- close format file so as not to affect READ's
SET FORMAT TO

DEACTIVATE WINDOW print
RESTORE WINDOW name FROM name.win
ACTIVATE WINDOW main
RESTORE WINDOW main FROM main2.win
IF .NOT. gl_batch
    DO AftAct
ENDIF

*-- Land use, vegetation & urban centers
CASE BAR() = 2
*-- Open Item level view/database and indexes
CLOSE DATABASES
lc_dbf='Y'
lc_message="0"

```

```

ON ERROR lc_message=LTRIM(STR(ERROR()))+" "+MESSAGE()
SET VIEW TO C_S_LUSE.QBE
ON ERROR
gn_error=VAL(lc_message)
IF gn_error > 0
DO Pause WITH "Error opening C_S_LUSE.QBE"
gn_error=0
lc_file="SET"+gc_prognum
DO &lc_file.
RETURN
ENDIF
lc_new='y'
SAVE WINDOW main TO main2
SET STATUS OFF
SET ORDER TO serial
GOTO TOP
IF .NOT. SEEK(serialno)
ACTIVATE WINDOW alarm
DO BELL1000
@ 0,0 SAY "No objective information."
mnkey=INKEY(3)
DO BELL1000
DEACTIVATE WINDOW alarm
RETURN
ENDIF
ACTIVATE WINDOW print
@ 10,32 SAY "Please wait..."
IF .NOT. gl_batch
DO BefAct
ENDIF
SET MESSAGE TO
*-- Desc: attach format file C_S_LUSE
SET FORMAT TO C_S_LUSE
EDIT FOR SERIAL=serialno NOFOLLOW NOAPPEND NOMENU NODELETE NOEDIT
*-- close format file so as not to affect READ's
SET FORMAT TO

DEACTIVATE WINDOW print
RESTORE WINDOW name FROM name.win
ACTIVATE WINDOW main
RESTORE WINDOW main FROM main2
IF .NOT. gl_batch
DO AftAct
ENDIF

*-- Development plans
CASE BAR() = 3
*-- Open Item level view/database and indexes
CLOSE DATABASES
lc_dbf='y'
lc_message="0"
ON ERROR lc_message=LTRIM(STR(ERROR()))+" "+MESSAGE()
SET VIEW TO C_S_DLOP.QBE
ON ERROR
gn_error=VAL(lc_message)
IF gn_error > 0
DO Pause WITH "Error opening C_S_DLOP.QBE"
gn_error=0
lc_file="SET"+gc_prognum
DO &lc_file.
RETURN
ENDIF
lc_new='y'
SAVE WINDOW main TO main2
SET STATUS OFF
SET ORDER TO serial
GOTO TOP
IF .NOT. SEEK(serialno)
ACTIVATE WINDOW alarm
DO BELL1000
@ 0,0 SAY "No objective information."
mnkey=INKEY(3)
DO BELL1000
DEACTIVATE WINDOW alarm
RETURN
ENDIF
ACTIVATE WINDOW print
@ 10,32 SAY "Please wait..."
IF .NOT. gl_batch
DO BefAct
ENDIF
SET MESSAGE TO
*-- Desc: attach format file C_S_DLOP
SET FORMAT TO C_S_DLOP
EDIT FOR SERIAL=serialno NOFOLLOW NOAPPEND NOMENU NODELETE NOEDIT

```

```

*-- close format file so as not to affect READ's
SET FORMAT TO

DEACTIVATE WINDOW print
RESTORE WINDOW name FROM name.win
ACTIVATE WINDOW main
RESTORE WINDOW main FROM main2.win
IF .NOT. gl batch
  DO AftAct
ENDIF

*-- Fauna/flora
CASE BAR() = 4
*-- Open Item level view/database and indexes
CLOSE DATABASES
lc dbf='Y'
lc message="0"
ON ERROR lc message=LTRIM(STR(ERROR()))+" "+MESSAGE()
SET VIEW TO C_S_FLOR.QBE
ON ERROR
gn_error=VAL(lc_message)
IF gn_error > 0
  DO Pause WITH "Error opening C_S_FLOR.QBE"
  gn_error=0
  lc file="SET"+gc_prognum
  DO &lc file.
  RETURN
ENDIF
lc new='Y'
SAVE WINDOW main TO main2
SET ORDER TO serial
GOTO TOP
IF .NOT. SEEK(serialno)
  ACTIVATE WINDOW alarm
  DO BELL1000
  @ 0,0 SAY "No objective information."
  mnkey=INKEY(3)
  DO BELL1000
  DEACTIVATE WINDOW alarm
  RETURN
ENDIF
SET STATUS OFF
ACTIVATE WINDOW print
@ 10,32 SAY "Please wait..."
IF .NOT. gl batch
  DO BefAct
ENDIF
SET MESSAGE TO
*-- Desc: attach format file C_S_FLOR
SET FORMAT TO C_S_FLOR
EDIT FOR SERIAL=serialno NOFOLLOW NOAPPEND NOMENU NODELETE NOEDIT
*-- close format file so as not to affect READ's
SET FORMAT TO

DEACTIVATE WINDOW print
RESTORE WINDOW name FROM name.win
ACTIVATE WINDOW main
RESTORE WINDOW main FROM main2.win
IF .NOT. gl batch
  DO AftAct
ENDIF

*-- Activities and others
CASE BAR() = 5
*-- Open Item level view/database and indexes
CLOSE DATABASES
lc dbf='Y'
lc message="0"
ON ERROR lc message=LTRIM(STR(ERROR()))+" "+MESSAGE()
SET VIEW TO C_S_ACT.QBE
ON ERROR
gn_error=VAL(lc_message)
IF gn_error > 0
  DO Pause WITH "Error opening C_S_ACT.QBE"
  gn_error=0
  lc file="SET"+gc_prognum
  DO &lc file.
  RETURN
ENDIF
lc new='Y'
SAVE WINDOW main TO main2
SET STATUS OFF
ACTIVATE WINDOW print
@ 10,32 SAY "Please wait..."

```



```

IF .NOT. gl_batch
  DO BefAct
ENDIF
SET MESSAGE TO
*-- Desc: attach format file C_S_ACT
SET FORMAT TO C S ACT
EDIT FOR SERIAL=serialno NOFOLLOW NOAPPEND NOMENU NODELETE NOEDIT
*-- close format file so as not to affect READ's
SET FORMAT TO

DEACTIVATE WINDOW print
RESTORE WINDOW name FROM name.win
ACTIVATE WINDOW main
RESTORE WINDOW main FROM main2.win
IF .NOT. gl_batch
  DO AftAct
ENDIF
ENDCASE

SET MESSAGE TO
IF gc quit='Q'
  IF LEFT(entryflg,1) = "B"
    DEACTIVATE MENU
  ELSE
    DEACTIVATE POPUP && C_SOCIO
  ENDIF
ENDIF
IF lc new='Y'
  lc_file="SET"+gc_prognum
  DO &lc_file.
ENDIF
IF lc dbf='Y' .AND. .NOT. lc_new='Y'
  lc_file="DBF"+gc_prognum
  DO &lc_file.
ENDIF
RETURN
*-- EOP: ACT011 - Menu C_SOCIO

PROCEDURE B MOUTH
PARAMETER entryflg
*-----
* Program.....: B MOUTH.PRG
* Author.....: K.Sasabe, National River Mouths Study
* Date.....: 8-27-93
* Notice.....: Japan International Cooperation Agency (JICA)
* Generated by.: dBASE IV version 1.5
* Description...: Menu actions
*-----
PRIVATE gc_prognum
gc_prognum="012"

DO SET012
IF gn_error > 0
  gn_error=0
RETURN
ENDIF

ACTIVATE POPUP B_MOUTH

RETURN
*-- EOP: B_MOUTH

PROCEDURE SET012
*-----
* B MOUTH setup routine. This procedure will open up the file or
* view for this menu, and setup the color scheme.
*-----

ON KEY LABEL F1 DO IHELP1          && Define F1-Help key

DO DBF012                          && open menu level database

IF gn_error = 0
  IF gl_color .AND. .NOT. SET("ATTRIBUTES") = ;
    "W/B,RG+/GB,N/N "+CHR(38)+CHR(38)+" RG+/N,W/B,RG+/GB,B/W,N/GB"
    SET COLOR OF NORMAL TO W+/B
    SET COLOR OF MESSAGES TO RG+/N
    SET COLOR OF TITLES TO W/B
    SET COLOR OF HIGHLIGHT TO RG+/GB
    SET COLOR OF BOX TO RG+/GB
    SET COLOR OF INFORMATION TO B/W
    SET COLOR OF FIELDS TO N/GB
  ENDIF
ENDIF

```

```

RETURN
*-- EOP: SET012 - Menu B_MOUTH

PROCEDURE DBF012
*-----
* B_MOUTH open DBF or view procedure
*-----
PRIVATE lc_message
CLOSE DATABASES
*-- Open menu level view/database
lc_message="0"
ON ERROR lc_message=LTRIM(STR(ERROR()))+" "+MESSAGE()
USE RESEAR_1.DBF
IF "" <> DBF()
SET INDEX TO RESEAR_1.MDX
ENDIF
SET ORDER TO SERIAL
ON ERROR
gn_error=VAL(lc_message)
IF gn_error > 0
DO Pause WITH "Error opening RESEAR_1.DBF or index(es) RESEAR_1.MDX"
lc_new='Y'
RETURN
ENDIF
lc_new='Y'
RETURN
*-- EOP: DBF012 - Menu B_MOUTH

PROCEDURE ACT012
*-----
* Begin B MOUTH: POPUP Menu Actions.
* Includes before item, action, and after item as needed
*-----
PRIVATE lc_new, lc_dbf
lc_new=' '
lc_dbf=' '
gc_scope=''
DO CASE

*-- Geomorphology
CASE BAR() = 1
*-- Open Item level view/database and indexes
CLOSE DATABASES
lc_dbf='Y'
lc_message="0"
ON ERROR lc_message=LTRIM(STR(ERROR()))+" "+MESSAGE()
SET VIEW TO C_M_GEO.QBE
ON ERROR
gn_error=VAL(lc_message)
IF gn_error > 0
DO Pause WITH "Error opening C_M_GEO.QBE"
gn_error=0
lc_file="SET"+gc_prognum
DO &lc_file.
RETURN
ENDIF
lc_new='Y'
SAVE WINDOW main TO main2.win
ACTIVATE WINDOW alarm
mok="Y"
@ 0,0 SAY "Printer ready? (Y/N):" GET mok
READ
DEACTIVATE WINDOW alarm
pl=68
pitch="&17C" && 7*1/72 increments
DO P SET WITH mok,pl,pitch
ACTIVATE WINDOW print
IF .NOT. gl_batch
DO BefAct
ENDIF
SET MESSAGE TO
*-- Desc: Report
gn_pkey = 0
DO PrintSet
IF gn_pkey <> 27 && esc
REPORT FORM B_M_GEO NOEJECT &gc_scope.
DO Cleanup
ENDIF

DO P RESET WITH mok
DEACTIVATE WINDOW print
ACTIVATE WINDOW main
RESTORE WINDOW main FROM main2.win
IF .NOT. gl_batch

```

```

DO AftAct
ENDIF

*-- Classification
CASE BAR() = 2
*-- Open Item level view/database and indexes
CLOSE DATABASES
lc dbf='Y'
lc message="0"
ON ERROR lc message=LTRIM(STR(ERROR()))+" "+MESSAGE()
SET VIEW TO ADD_INFO.QBE
ON ERROR
gn_error=VAL(lc message)
IF gn_error > 0
DO Pause WITH "Error opening ADD_INFO.QBE"
gn_error=0
lc file="SET"+gc_prognum
DO &lc file.
RETURN
ENDIF
lc new='Y'
SAVE WINDOW main TO main2.win
ACTIVATE WINDOW alarm
mok="Y"
@ 0,0 SAY "Printer ready? (Y/N):" GET mok
READ
DEACTIVATE WINDOW alarm
pl=75
pitch="&16.52C"
DO P SET WITH mok,pl,pitch
ACTIVATE WINDOW print
IF .NOT. gl batch
DO BefAct
ENDIF
SET MESSAGE TO
*-- Desc: Report
gn_pkey = 0
DO PrintSet
IF gn_pkey <> 27 && esc
REPORT FORM B_MCLASS NOEJECT &gc_scope.
DO Cleanup
ENDIF

DO P RESET WITH mok
DEACTIVATE WINDOW print
ACTIVATE WINDOW main
RESTORE WINDOW main FROM main2.win
IF .NOT. gl batch
DO AftAct
ENDIF

*-- Oceanographical data
CASE BAR() = 3
*-- Open Item level view/database and indexes
CLOSE DATABASES
lc dbf='Y'
lc message="0"
ON ERROR lc message=LTRIM(STR(ERROR()))+" "+MESSAGE()
SET VIEW TO C_M_OCEA.QBE
ON ERROR
gn_error=VAL(lc message)
IF gn_error > 0
DO Pause WITH "Error opening C_M_OCEA.QBE"
gn_error=0
lc file="SET"+gc_prognum
DO &lc file.
RETURN
ENDIF
lc new='Y'
SAVE WINDOW main TO main2.win
ACTIVATE WINDOW alarm
mok="Y"
@ 0,0 SAY "Printer ready? (Y/N):" GET mok
READ
DEACTIVATE WINDOW alarm
pl=63
pitch="&17.56C"
DO P SET WITH mok,pl,pitch
ACTIVATE WINDOW print
IF .NOT. gl batch
DO BefAct
ENDIF
SET MESSAGE TO
*-- Desc: Report
gn_pkey = 0

```

```

DO PrintSet
IF gn_pkey <> 27 && esc
  REPORT FORM B_M_OCEA NOEJECT &gc_scope.
DO Cleanup
ENDIF

DO P RESET WITH mok
DEACTIVATE WINDOW print
ACTIVATE WINDOW main
RESTORE WINDOW main FROM main2.win
IF .NOT. gl_batch
  DO AftAct
ENDIF

*-- Structures
CASE BAR() = 4
*-- Open Item level view/database and indexes
CLOSE DATABASES
lc_dbf='Y'
lc_message="0"
ON ERROR lc_message=LTRIM(STR(ERROR()))+" "+MESSAGE()
SET VIEW TO C_M_STR1.QBE
ON ERROR
gn_error=VAL(lc_message)
IF gn_error > 0
  DO Pause WITH "Error opening C_M_STR1.QBE"
  gn_error=0
  lc_file="SET"+gc_prognum
  DO &lc_file.
  RETURN
ENDIF
lc_new='Y'
SAVE WINDOW main TO main2.win
ACTIVATE WINDOW alarm
mok="Y"
@ 0,0 SAY "Printer ready? (Y/N):" GET mok
READ
DEACTIVATE WINDOW alarm
pl=66
pitch="&17.2C"
DO P SET WITH mok,pl,pitch
ACTIVATE WINDOW print
IF .NOT. gl_batch
  DO BefAct
ENDIF
SET MESSAGE TO
*-- Desc: Report
gn_pkey = 0
DO PrintSet
IF gn_pkey <> 27 && esc
  REPORT FORM B_M_STR NOEJECT &gc_scope.
DO Cleanup
ENDIF

DO P RESET WITH mok
DEACTIVATE WINDOW print
ACTIVATE WINDOW main
RESTORE WINDOW main FROM main2.win
IF .NOT. gl_batch
  DO AftAct
ENDIF
ENDCASE

SET MESSAGE TO
IF gc_quit='Q'
  IF LEFT(entryflg,1) = "B"
    DEACTIVATE MENU
  ELSE
    DEACTIVATE POPUP && B_MOUTH
  ENDIF
ENDIF
IF lc_new='Y'
  lc_file="SET"+gc_prognum
  DO &lc_file.
ENDIF
IF lc_dbf='Y' .AND. .NOT. lc_new='Y'
  lc_file="DBF"+gc_prognum
  DO &lc_file.
ENDIF
RETURN
*-- EOP: ACT012 - Menu B_MOUTH

PROCEDURE B_SOCIO
PARAMETER entryflg

```

```

*-----
* Program.....: B_SOCIO.PRG
* Author.....: K.Sasabe, National River Mouths Study
* Date.....: 8-27-93
* Notice.....: Japan International Cooperation Agency (JICA)
* Generated by.: dBASE IV version 1.5
* Description.: Menu actions
*-----
PRIVATE gc_prognum
gc_prognum="013"

DO SET013
IF gn_error > 0
  gn_error=0
  RETURN
ENDIF

ACTIVATE POPUP B_SOCIO

RETURN
*-- EOP: B_SOCIO

PROCEDURE SET013
*-----
* B_SOCIO setup routine. This procedure will open up the file or
* view for this menu, and setup the color scheme.
*-----
ON KEY LABEL F1 DO IHELP1          && Define F1-Help key

DO DBF013                          && open menu level database

IF gn_error = 0
  IF ! color AND .NOT. SET("ATTRIBUTES") = ;
    "W/B,RG+/GB,N/N "+CHR(38)+CHR(38)+" RG+/N,W/B,RG+/GB,B/W,N/GB"
    SET COLOR OF NORMAL TO W+/B
    SET COLOR OF MESSAGES TO RG+/N
    SET COLOR OF TITLES TO W/B
    SET COLOR OF HIGHLIGHT TO RG+/GB
    SET COLOR OF BOX TO RG+/GB
    SET COLOR OF INFORMATION TO B/W
    SET COLOR OF FIELDS TO N/GB
  ENDIF
ENDIF

RETURN
*-- EOP: SET013 - Menu B_SOCIO

PROCEDURE DBF013
*-----
* B_SOCIO open DBF or view procedure
*-----
PRIVATE lc_message
CLOSE DATABASES
*-- Open menu level view/database
lc_message="0"
ON ERROR lc_message=LTRIM(STR(ERROR()))+" "+MESSAGE()
USE RESEAR_1.DBF
IF "" <> DBF()
  SET INDEX TO RESEAR_1.MDX
ENDIF
SET ORDER TO SERIAL
ON ERROR
gn_error=VAL(lc_message)
IF gn_error > 0
  DO Pause WITH "Error opening RESEAR_1.DBF or index(es) RESEAR_1.MDX"
  lc_new='Y'
  RETURN
ENDIF
lc_new='Y'
RETURN
*-- EOP: DBF013 - Menu B_SOCIO

PROCEDURE ACT013
*-----
* Begin B_SOCIO: POPUP Menu Actions.
* Includes before item, action, and after item as needed
*-----
PRIVATE lc_new, lc_dbf
lc_new=' '
lc_dbf=' '
gc_scope=' '
DO CASE

  *-- Land Use

```

```

CASE BAR() = 1
*-- Open Item level view/database and indexes
CLOSE DATABASES
lc dbf='Y'
lc message="0"
ON ERROR lc message=LTRIM(STR(ERROR()))+" "+MESSAGE()
SET VIEW TO ADD_INFO.QBE
ON ERROR
gn error=VAL(lc message)
IF gn error > 0
    DO Pause WITH "Error opening ADD_INFO.QBE"
    gn error=0
    lc file="SET"+gc_prognum
    DO &lc file.
    RETURN
ENDIF
lc new='Y'
SAVE WINDOW main TO main2.win
ACTIVATE WINDOW alarm
mok="Y"
@ 0,0 SAY "Printer ready? (Y/N):" GET mok
READ
DEACTIVATE WINDOW alarm
pl=63
pitch="&17.56C"
DO P SET WITH mok,pl,pitch
ACTIVATE WINDOW print
IF .NOT. gl_batch
    DO BefAct
ENDIF
SET MESSAGE TO
*-- Desc: Report
gn_pkey = 0
DO PrintSet
IF gn_pkey <> 27 && esc
    REPORT FORM B_S_LUSE NOEJECT &gc_scope.
    DO Cleanup
ENDIF

DO P RESET WITH mok
DEACTIVATE WINDOW print
ACTIVATE WINDOW main
RESTORE WINDOW main FROM main2.win
IF .NOT. gl_batch
    DO AftAct
ENDIF

*-- Major Urban Centers
CASE BAR() = 2
*-- Open Item level view/database and indexes
CLOSE DATABASES
lc dbf='Y'
lc message="0"
ON ERROR lc message=LTRIM(STR(ERROR()))+" "+MESSAGE()
SET VIEW TO C_S_LUSE.QBE
ON ERROR
gn error=VAL(lc message)
IF gn error > 0
    DO Pause WITH "Error opening C_S_LUSE.QBE"
    gn error=0
    lc file="SET"+gc_prognum
    DO &lc file.
    RETURN
ENDIF
lc new='Y'
SAVE WINDOW main TO main2.win
ACTIVATE WINDOW alarm
mok="Y"
@ 0,0 SAY "Printer ready? (Y/N):" GET mok
READ
DEACTIVATE WINDOW alarm
pl=61
pitch="&16D"
DO P SET WITH mok,pl,pitch
ACTIVATE WINDOW print
IF .NOT. gl_batch
    DO BefAct
ENDIF
SET MESSAGE TO
*-- Desc: Report
gn_pkey = 0
DO PrintSet
IF gn_pkey <> 27 && esc
    REPORT FORM B_S_URBN NOEJECT &gc_scope.
    DO Cleanup

```

```

ENDIF

DO P RESET WITH mok
DEACTIVATE WINDOW print
ACTIVATE WINDOW main
RESTORE WINDOW main FROM main2.win
IF .NOT. gl_batch
DO AttAct
ENDIF
ENDCASE

SET MESSAGE TO
IF gc_quit='Q'
IF LEFT(entryflg,1) = "B"
DEACTIVATE MENU
ELSE
DEACTIVATE POPUP && B_SOCIO
ENDIF
ENDIF
IF lc_new='Y'
lc_file="SET"+gc_prognum
DO &lc_file.
ENDIF
IF lc_dbf='Y' .AND. .NOT. lc_new='Y'
lc_file="DBF"+gc_prognum
DO &lc_file.
ENDIF
RETURN
*-- EOP: ACT013 - Menu B_SOCIO

PROCEDURE RESEAR_1
PARAMETER entryflg
*-----
* Program.....: RESEAR_1.PRG
* Author.....: K.Sasabe, National River Mouths Study
* Date.....: 8-27-93
* Notice.....: Japan International Cooperation Agency (JICA)
* Generated by.: dBASE IV version 1.5
* Description.: Menu actions
*-----
PRIVATE gc_prognum
gc_prognum="014"

DO SET014
IF gn_error > 0
gn_error=0
RETURN
ENDIF

ACTIVATE POPUP RESEAR_1

RETURN
*-- EOP: RESEAR_1

PROCEDURE SET014
*-----
* RESEAR_1 setup routine. This procedure will open up the file or
* view for this menu, and setup the color scheme.
*-----

ON KEY LABEL F1 DO IHHELP1          && Define F1-Help key

DO DBF014                            && open menu level database

IF gn_error = 0
IF gl_color .AND. .NOT. SET("ATTRIBUTES") = ;
" W/B,RG+/GB,N/N "+CHR(38)+CHR(38)+" RG+/N,W/B,RG+/GB,B/W,N/GB"
SET COLOR OF NORMAL TO W+/B
SET COLOR OF MESSAGES TO RG+/N
SET COLOR OF TITLES TO W/B
SET COLOR OF HIGHLIGHT TO RG+/GB
SET COLOR OF BOX TO RG+/GB
SET COLOR OF INFORMATION TO B/W
SET COLOR OF FIELDS TO N/GB
ENDIF
ENDIF

RETURN
*-- EOP: SET014 - Menu RESEAR_1

PROCEDURE DBF014
*-----
* RESEAR_1 open DBF or view procedure
*-----
PRIVATE lc_message

```

```

CLOSE DATABASES
*-- Open menu level view/database
lc message="0"
ON ERROR lc message=LTRIM(STR(ERROR()))+" "+MESSAGE()
USE RESEAR_1.DBF
IF "" <> DBF()
  SET INDEX TO RESEAR_1.MDX
ENDIF
SET ORDER TO SERIAL
ON ERROR
gn error=VAL(lc message)
IF gn error > 0
  DO Pause WITH "Error opening RESEAR_1.DBF or index(es) RESEAR_1.MDX"
  lc new='Y'
  RETURN
ENDIF
lc new='Y'
RETURN
*-- EOP: DBF014 - Menu RESEAR_1

```

PROCEDURE ACT014

```

*-----
* Begin RESEAR_1: POPUP Menu Actions.
* Includes before item, action, and after item as needed
*-----

```

```

PRIVATE lc_new, lc_dbf
lc_new=' '
lc_dbf=' '
gc_scope=''
DO CASE

```

```

*-- Part 1/3
CASE BAR() = 1
  SAVE WINDOW main TO main3
  SET STATUS ON
  ACTIVATE WINDOW print
  IF .NOT. gl_batch
    DO BefAct
  ENDIF
  SET MESSAGE TO
  *-- Desc: attach format file RES_1_1
  SET FORMAT TO RES_1_1
  EDIT &gc scope. NOFOLLOW NOAPPEND NODELETE
  *-- close format file so as not to affect READ's
  SET FORMAT TO

```

```

  DEACTIVATE WINDOW print
  SET STATUS OFF
  ACTIVATE WINDOW main
  RESTORE WINDOW main FROM main3
  IF .NOT. gl_batch
    DO AftAct
  ENDIF

```

```

*-- Part 2/3
CASE BAR() = 2
  SAVE WINDOW main TO main3
  SET STATUS ON
  ACTIVATE WINDOW print
  IF .NOT. gl_batch
    DO BefAct
  ENDIF
  SET MESSAGE TO
  *-- Desc: attach format file RES_1_2
  SET FORMAT TO RES_1_2
  EDIT &gc scope. NOFOLLOW NOAPPEND NODELETE
  *-- close format file so as not to affect READ's
  SET FORMAT TO

```

```

  DEACTIVATE WINDOW print
  SET STATUS OFF
  ACTIVATE WINDOW main
  RESTORE WINDOW main FROM main3
  IF .NOT. gl_batch
    DO AftAct
  ENDIF

```

```

*-- Part 3/3
CASE BAR() = 3
  SAVE WINDOW main TO main3
  SET STATUS ON
  ACTIVATE WINDOW print
  IF .NOT. gl_batch

```



```

    DO BefAct
  ENDF
  SET MESSAGE TO
  *-- Desc: attach format file RES_1_3
  SET FORMAT TO RES_1_3
  EDIT &gc scope. NOFOLLOW NOAPPEND NODELETE
  *-- close format file so as not to affect READ's
  SET FORMAT TO

  DEACTIVATE WINDOW print
  SET STATUS OFF
  ACTIVATE WINDOW main
  RESTORE WINDOW main FROM main3
  IF .NOT. gl_batch
    DO AftAct
  ENDF
ENDCASE

SET MESSAGE TO
IF gc_quit='Q'
  IF LEFT(entryflg,1) = "B"
    DEACTIVATE MENU
  ELSE
    DEACTIVATE POPUP && RESEAR_1
  ENDF
ENDIF
IF lc_new='Y'
  lc_file="SET"+gc_prognum
  DO &lc_file.
ENDIF
RETURN
*-- EOP: ACT014 - Menu RESEAR_1

PROCEDURE RESEAR_2
PARAMETER entryflg
*-----
* Program.....: RESEAR_2.PRG
* Author.....: K.Sasabe, National River Mouths Study
* Date.....: 8-27-93
* Notice.....: Japan International Cooperation Agency (JICA)
* Generated by.: dBASE IV version 1.5
* Description..: Menu actions
*-----
PRIVATE gc_prognum
gc_prognum="015"

DO SET015
IF gn_error > 0
  gn_error=0
RETURN
ENDIF

ACTIVATE POPUP RESEAR_2

RETURN
*-- EOP: RESEAR_2

PROCEDURE SET015
*-----
* RESEAR_2 setup routine. This procedure will open up the file or
* view for this menu, and setup the color scheme.
*-----

ON KEY LABEL F1 DO !HELP1          && Define F1-Help key

DO DBF015                          && open menu level database

IF gn_error = 0
  IF gl_color .AND. .NOT. SET("ATTRIBUTES") = ;
    "W/B,RG+/GB,N/N "+CHR(38)+CHR(38)+" RG+/N,W/B,RG+/GB,B/W,N/GB"
    SET COLOR OF NORMAL TO W+/B
    SET COLOR OF MESSAGES TO RG+/N
    SET COLOR OF TITLES TO W/B
    SET COLOR OF HIGHLIGHT TO RG+/GB
    SET COLOR OF BOX TO RG+/GB
    SET COLOR OF INFORMATION TO B/W
    SET COLOR OF FIELDS TO N/GB
  ENDF
ENDIF

RETURN
*-- EOP: SET015 - Menu RESEAR_2

PROCEDURE DBF015
*-----

```

```

* RESEAR_2 open DBF or view procedure
*-----
PRIVATE lc_message
CLOSE DATABASES
*-- Open menu level view/database
lc_message="0"
ON ERROR lc_message=LTRIM(STR(ERROR()))+" "+MESSAGE()
USE RESEAR_1.DBF
IF "" <> DBF()
    SET INDEX TO RESEAR_1.MDX
ENDIF
SET ORDER TO SERIAL
ON ERROR
gn_error=VAL(lc_message)
IF gn_error > 0
    DO Pause WITH "Error opening RESEAR_1.DBF or index(es) RESEAR_1.MDX"
    lc_new='Y'
    RETURN
ENDIF
lc_new='Y'
RETURN
*-- EOP: DBF015 - Menu RESEAR_2

PROCEDURE ACT015
*-----
* Begin RESEAR 2: POPUP Menu Actions.
* Includes before item, action, and after item as needed
*
*-----
PRIVATE lc_new, lc_dbf
lc_new=' '
lc_dbf=' '
gc_scope=''
DO CASE

*-- Part 1/3
CASE BAR() = 1
*-- Open Item level view/database and indexes
CLOSE DATABASES
lc_dbf='Y'
lc_message="0"
ON ERROR lc_message=LTRIM(STR(ERROR()))+" "+MESSAGE()
SET VIEW TO RESEAR_2.QBE
ON ERROR
gn_error=VAL(lc_message)
IF gn_error > 0
    DO Pause WITH "Error opening RESEAR_2.QBE"
    gn_error=0
    lc_file="SET"+gc_prognum
    DO &lc_file.
    RETURN
ENDIF
lc_new='Y'
SAVE WINDOW main TO main3
SET STATUS ON
ACTIVATE WINDOW print
IF .NOT. gl_batch
    DO BefAct
ENDIF
SET MESSAGE TO
*-- Desc: attach format file RES_2_1
SET FORMAT TO RES_2_1
EDIT &gc_scope, NOFOLLOW NOAPPEND NODELETE
*-- close format file so as not to affect READ's
SET FORMAT TO

DEACTIVATE WINDOW print
SET STATUS OFF
ACTIVATE WINDOW main
RESTORE WINDOW main FROM main3
IF .NOT. gl_batch
    DO AftAct
ENDIF

*-- Part 2/3
CASE BAR() = 2
*-- Open Item level view/database and indexes
CLOSE DATABASES
lc_dbf='Y'
lc_message="0"
ON ERROR lc_message=LTRIM(STR(ERROR()))+" "+MESSAGE()
SET VIEW TO RESEAR_2.QBE
ON ERROR
gn_error=VAL(lc_message)
IF gn_error > 0

```

```

DO Pause WITH "Error opening RESEAR_2.QBE"
gn_error=0
lc_file="SET"+gc_prognum
DO &lc_file.
RETURN
ENDIF
lc_new='Y'
SAVE WINDOW main TO main3
SET STATUS ON
ACTIVATE WINDOW print
IF .NOT. gl_batch
DO BefAct
ENDIF
SET MESSAGE TO
*-- Desc: attach format file RES_2_2
SET FORMAT TO RES_2_2
EDIT &gc_scope. NOFOLLOW NOAPPEND NODELETE
*-- close format file so as not to affect READ's
SET FORMAT TO

DEACTIVATE WINDOW print
SET STATUS OFF
ACTIVATE WINDOW main
RESTORE WINDOW main FROM main3
IF .NOT. gl_batch
DO AftAct
ENDIF

*-- Part 3/3
CASE BAR() = 3
*-- Open Item level view/database and indexes
CLOSE DATABASES
lc_dbf='Y'
lc_message="0"
ON ERROR lc_message=LTRIM(STR(ERROR()))+" "+MESSAGE()
SET VIEW TO RESEAR_2.QBE
ON ERROR
gn_error=VAL(lc_message)
IF gn_error > 0
DO Pause WITH "Error opening RESEAR_2.QBE"
gn_error=0
lc_file="SET"+gc_prognum
DO &lc_file.
RETURN
ENDIF
lc_new='Y'
SAVE WINDOW main TO main3
SET STATUS ON
ACTIVATE WINDOW print
IF .NOT. gl_batch
DO BefAct
ENDIF
SET MESSAGE TO
*-- Desc: attach format file RES_2_3
SET FORMAT TO RES_2_3
EDIT &gc_scope. NOFOLLOW NOAPPEND NODELETE
*-- close format file so as not to affect READ's
SET FORMAT TO

DEACTIVATE WINDOW print
SET STATUS OFF
ACTIVATE WINDOW main
RESTORE WINDOW main FROM main3
IF .NOT. gl_batch
DO AftAct
ENDIF
ENDCASE

SET MESSAGE TO
IF gc_quit='Q'
IF LEFT(entryflg,1) = "B"
DEACTIVATE MENU
ELSE
DEACTIVATE POPUP && RESEAR_2
ENDIF
ENDIF
IF lc_new='Y'
lc_file="SET"+gc_prognum
DO &lc_file.
ENDIF
IF lc_dbf='Y' .AND. .NOT. lc_new='Y'
lc_file="DBF"+gc_prognum
DO &lc_file.
ENDIF

```

```

RETURN
*-- EOP: ACT015 - Menu RESEAR_2

PROCEDURE RESEAR_3
PARAMETER entryfTg
*-----
* Program.....: RESEAR_3.PRG
* Author.....: K.SasaBe, National River Mouths Study
* Date.....: 8-27-93
* Notice.....: Japan International Cooperation Agency (JICA)
* Generated by.: dBASE IV version 1.5
* Description..: Menu actions
*-----
PRIVATE gc_prognum
gc_prognum="016"

DO SET016
IF gn_error > 0
  gn_error=0
  RETURN
ENDIF

ACTIVATE POPUP RESEAR_3

RETURN
*-- EOP: RESEAR_3

PROCEDURE SET016
*-----
* RESEAR_3 setup routine. This procedure will open up the file or
* view for this menu, and setup the color scheme.
*-----
ON KEY LABEL F1 DO IHHELP1          && Define F1-Help key
DO DBF016                            && open menu level database

IF gn_error = 0
  IF q1 color .AND. .NOT. SET("ATTRIBUTES") = ;
    "W+/B, RG+/GB, N/N "+CHR(38)+CHR(38)+" RG+/N, W/B, RG+/GB, B/W, N/GB"
    SET COLOR OF NORMAL TO W+/B
    SET COLOR OF MESSAGES TO RG+/N
    SET COLOR OF TITLES TO W/B
    SET COLOR OF HIGHLIGHT TO RG+/GB
    SET COLOR OF BOX TO RG+/GB
    SET COLOR OF INFORMATION TO B/W
    SET COLOR OF FIELDS TO N/GB
  ENDIF
ENDIF

RETURN
*-- EOP: SET016 - Menu RESEAR_3

PROCEDURE DBF016
*-----
* RESEAR_3 open DBF or view procedure
*-----
PRIVATE lc_message
CLOSE DATABASES
*-- Open menu level view/database
lc_message="0"
ON ERROR lc_message=LTRIM(STR(ERROR()))+" "+MESSAGE()
USE RESEAR_1.DBF
IF "" <> DBF()
  SET INDEX TO RESEAR_1.MDX
ENDIF
SET ORDER TO SERIAL
ON ERROR
gn_error=VAL(lc_message)
IF gn_error > 0
  DO Pause WITH "Error opening RESEAR_1.DBF or index(es) RESEAR_1.MDX"
  lc_new='Y'
  RETURN
ENDIF
lc_new='Y'

RETURN
*-- EOP: DBF016 - Menu RESEAR_3

PROCEDURE ACT016
*-----
* Begin RESEAR_3: POPUP Menu Actions.
* Includes before item, action, and after item as needed
*-----
PRIVATE lc_new, lc_dbf
lc_new=' '

```

```

lc dbf=' '
gc scope=''
DO CASE

*-- Part 1/4
CASE BAR() = 1
*-- Open Item level view/database and indexes
CLOSE DATABASES
lc dbf='Y'
lc message="0"
ON ERROR lc message=LTRIM(STR(ERROR()))+" "+MESSAGE()
SET VIEW TO RESEAR_3.QBE
ON ERROR
gn_error=VAL(lc_message)
IF gn_error > 0
DO Pause WITH "Error opening RESEAR_3.QBE"
gn_error=0
lc file="SET"+gc_prognum
DO &lc file.
RETURN
ENDIF
lc new='Y'
SAVE WINDOW main TO main3
SET STATUS ON
ACTIVATE WINDOW print
IF .NOT. gl_batch
DO BefAct
ENDIF
SET MESSAGE TO
*-- Desc: attach format file RES_3_1
SET FORMAT TO RES_3_1
EDIT &gc_scope. NOFOLLOW NOAPPEND NODELETE
*-- close format file so as not to affect READ's
SET FORMAT TO

DEACTIVATE WINDOW print
SET STATUS OFF
ACTIVATE WINDOW main
RESTORE WINDOW main FROM main3
IF .NOT. gl_batch
DO AftAct
ENDIF

*-- Part 2/4
CASE BAR() = 2
*-- Open Item level view/database and indexes
CLOSE DATABASES
lc dbf='Y'
lc message="0"
ON ERROR lc message=LTRIM(STR(ERROR()))+" "+MESSAGE()
SET VIEW TO RESEAR_3.QBE
ON ERROR
gn_error=VAL(lc_message)
IF gn_error > 0
DO Pause WITH "Error opening RESEAR_3.QBE"
gn_error=0
lc file="SET"+gc_prognum
DO &lc file.
RETURN
ENDIF
lc new='Y'
SAVE WINDOW main TO main3
SET STATUS ON
ACTIVATE WINDOW print
IF .NOT. gl_batch
DO BefAct
ENDIF
SET MESSAGE TO
*-- Desc: attach format file RES_3_2
SET FORMAT TO RES_3_2
EDIT &gc_scope. NOFOLLOW NOAPPEND NODELETE
*-- close format file so as not to affect READ's
SET FORMAT TO

DEACTIVATE WINDOW print
SET STATUS OFF
ACTIVATE WINDOW main
RESTORE WINDOW main FROM main3
IF .NOT. gl_batch
DO AftAct
ENDIF

*-- Part 3/4
CASE BAR() = 3

```

```

*-- Open Item level view/database and indexes
CLOSE DATABASES
lc dbf='Y'
lc_message="0"
ON ERROR lc_message=LTRIM(STR(ERROR()))+" "+MESSAGE()
SET VIEW TO RESEAR_3.QBE
ON ERROR
gn_error=VAL(lc_message)
IF gn_error > 0
  DO Pause WITH "Error opening RESEAR_3.QBE"
  gn_error=0
  lc_file="SET"+gc_prognum
  DO &lc_file.
  RETURN
ENDIF
lc_new='Y'
SAVE WINDOW main TO main3
SET STATUS ON
ACTIVATE WINDOW print
IF .NOT. gl_batch
  DO BefAct
ENDIF
SET MESSAGE TO
*-- Desc: attach format file RES_3_3
SET FORMAT TO RES 3 3
EDIT &gc_scope. NOFOLLOW NOAPPEND NODELETE
*-- close format file so as not to affect READ's
SET FORMAT TO

DEACTIVATE WINDOW print
SET STATUS OFF
ACTIVATE WINDOW main
RESTORE WINDOW main FROM main3
IF .NOT. gl_batch
  DO AftAct
ENDIF

*-- Part 4/4
CASE BAR() = 4
*-- Open Item level view/database and indexes
CLOSE DATABASES
lc dbf='Y'
lc_message="0"
ON ERROR lc_message=LTRIM(STR(ERROR()))+" "+MESSAGE()
SET VIEW TO RESEAR_3.QBE
ON ERROR
gn_error=VAL(lc_message)
IF gn_error > 0
  DO Pause WITH "Error opening RESEAR_3.QBE"
  gn_error=0
  lc_file="SET"+gc_prognum
  DO &lc_file.
  RETURN
ENDIF
lc_new='Y'
SAVE WINDOW main TO main3
SET STATUS ON
ACTIVATE WINDOW print
IF .NOT. gl_batch
  DO BefAct
ENDIF
SET MESSAGE TO
*-- Desc: attach format file RES_3_4
SET FORMAT TO RES 3 4
EDIT &gc_scope. NOFOLLOW NOAPPEND NODELETE
*-- close format file so as not to affect READ's
SET FORMAT TO

DEACTIVATE WINDOW print
SET STATUS OFF
ACTIVATE WINDOW main
RESTORE WINDOW main FROM main3
IF .NOT. gl_batch
  DO AftAct
ENDIF
ENDCASE

SET MESSAGE TO
IF gc_quit='Q'
  IF LEFT(entryflg,1) = "B"
    DEACTIVATE MENU
  ELSE
    DEACTIVATE POPUP && RESEAR_3
  ENDIF

```

```

ENDIF
IF lc_new='Y'
  lc_file="SET"+gc_prognum
  DO &lc_file.
ENDIF
IF lc_dbf='Y' .AND. .NOT. lc_new='Y'
  lc_file="DBF"+gc_prognum
  DO &lc_file.
ENDIF
RETURN
*-- EOP: ACT016 - Menu RESEAR_3

PROCEDURE C_M_STR
PARAMETER entryflg
*-----
* Program.....: C_M_STR.PRG
* Author.....: K.Sasabe, National River Mouths Study
* Date.....: 8-27-93
* Notice.....: Japan International Cooperation Agency (JICA)
* Generated by.: dBASE IV version 1.5
* Description..: Menu actions
*-----
PRIVATE gc_prognum
gc_prognum="017"

DO SET017
IF gn_error > 0
  gn_error=0
RETURN
ENDIF

ACTIVATE POPUP C_M_STR

RETURN
*-- EOP: C_M_STR

PROCEDURE SET017
*-----
* C_M_STR setup routine. This procedure will open up the file or
* view for this menu, and setup the color scheme.
*-----

ON KEY LABEL F1 DO IHELP1          && Define F1-Help key
DO DBF017                          && open menu level database

IF gn_error = 0
  IF ! color .AND. .NOT. SET("ATTRIBUTES") = ;
    "W+/B,RG+/GB,N/N "+CHR(38)+CHR(38)+" RG+/N,W/B,RG+/GB,B/W,N/GB"
    SET COLOR OF NORMAL TO W+/B
    SET COLOR OF MESSAGES TO RG+/N
    SET COLOR OF TITLES TO W/B
    SET COLOR OF HIGHLIGHT TO RG+/GB
    SET COLOR OF BOX TO RG+/GB
    SET COLOR OF INFORMATION TO B/W
    SET COLOR OF FIELDS TO N/GB
  ENDIF
ENDIF

RETURN
*-- EOP: SET017 - Menu C_M_STR

PROCEDURE DBF017
*-----
* C_M_STR open DBF or view procedure
*-----
PRIVATE lc_message
CLOSE DATABASES
*-- Open menu level view/database
lc_message="0"
ON ERROR lc_message=LTRIM(STR(ERROR()))+" "+MESSAGE()
USE RESEAR_1.DBF
IF "" <> DBF()
  SET INDEX TO RESEAR_1.MDX
ENDIF
SET ORDER TO SERIAL
ON ERROR
gn_error=VAL(lc_message)
IF gn_error > 0
  DO Pause WITH "Error opening RESEAR_1.DBF or index(es) RESEAR_1.MDX"
  lc_new='Y'
RETURN
ENDIF
lc_new='Y'
RETURN
*-- EOP: DBF017 - Menu C_M_STR

```

PROCEDURE ACT017

```

*-----
* Begin C M STR: POPUP Menu Actions.
* Includes Before item, action, and after item as needed
*-----
PRIVATE lc_new, lc_dbf
lc_new=' '
lc_dbf=' '
gc_scope=''
DO CASE

  *-- General information
  CASE BAR() = 1
    *-- Open Item level view/database and indexes
    CLOSE DATABASES
    lc_dbf='Y'
    lc_message="0"
    ON ERROR lc_message=LTRIM(STR(ERROR()))+" "+MESSAGE()
    SET VIEW TO C_M_STR1.QBE
    ON ERROR
    gn_error=VAL(lc_message)
    IF gn_error > 0
      DO Pause WITH "Error opening C_M_STR1.QBE"
      gn_error=0
      lc_file="SET"+gc_prognum
      DO &lc_file.
      RETURN
    ENDIF
    lc_new='Y'
    SAVE WINDOW main TO main3.win
    SET STATUS OFF
    ACTIVATE WINDOW print
    @ 10,32 SAY "Please wait..."
    IF .NOT. gl_batch
      DO BefAct
    ENDIF
    SET MESSAGE TO
    *-- Desc: attach format file C_M_STR1
    SET FORMAT TO C_M_STR1
    EDIT FOR SERIAL=serialno NOFOLLOW NOAPPEND NOMENU NODELETE NOEDIT
    *-- close format file so as not to affect READ's
    SET FORMAT TO

    DEACTIVATE WINDOW print
    RESTORE WINDOW name FROM name.win
    ACTIVATE WINDOW main
    RESTORE WINDOW main FROM main3.win
    IF .NOT. gl_batch
      DO AftAct
    ENDIF

  *-- Details
  CASE BAR() = 2
    *-- Open Item level view/database and indexes
    CLOSE DATABASES
    lc_dbf='Y'
    lc_message="0"
    ON ERROR lc_message=LTRIM(STR(ERROR()))+" "+MESSAGE()
    SET VIEW TO C_M_STR2.QBE
    ON ERROR
    gn_error=VAL(lc_message)
    IF gn_error > 0
      DO Pause WITH "Error opening C_M_STR2.QBE"
      gn_error=0
      lc_file="SET"+gc_prognum
      DO &lc_file.
      RETURN
    ENDIF
    lc_new='Y'
    SAVE WINDOW main TO main3
    SET ORDER TO serial
    GOTO TOP
    IF .NOT. SEEK(serialno)
      ACTIVATE WINDOW alarm
      DO BELL1000
      @ 0,0 SAY "No objective information."
      mnkey=INKEY(3)
      DO BELL1000
      DEACTIVATE WINDOW alarm
      RETURN
    ENDIF
    ACTIVATE WINDOW print
    @ 10,32 SAY "Please wait..."

```



```

IF .NOT. ql_batch
  DO BefAct
ENDIF
SET MESSAGE TO
*-- Desc: attach format file C_M_STR2
SET FORMAT TO C M STR2
EDIT FOR SERIAL=serialno NOFOLLOW NOAPPEND NOMENU NODELETE NOEDIT
*-- close format file so as not to affect READ's
SET FORMAT TO

DEACTIVATE WINDOW print
RESTORE WINDOW name FROM name.win
ACTIVATE WINDOW main
RESTORE WINDOW main FROM main3.win
IF .NOT. ql_batch
  DO AftAct
ENDIF
ENDCASE

SET MESSAGE TO
IF gc_quit='Q'
  IF LEFT(entryflg,1) = "B"
    DEACTIVATE MENU
  ELSE
    DEACTIVATE POPUP && C_M_STR
  ENDIF
ENDIF
IF lc_new='Y'
  lc_file="SET"+gc_prognum
  DO %lc_file.
ENDIF
IF lc_dbf='Y' .AND. .NOT. lc_new='Y'
  lc_file="DBF"+gc_prognum
  DO %lc_file.
ENDIF
RETURN
*-- EOP: ACT017 - Menu C_M_STR

```

```

*-----
* Program.....: R MOUTH.PRG
* Author.....: K. Sasabe, National River Mouths Study
* Date.....: 8-27-93
* Notice.....: Japan International Cooperation Agency (JICA)
* Generated by.: dBASE IV version 1.5
* Description..: Main routine for menu system
*-----

*-- Setup environment
SET CONSOLE OFF
IF TYPE("gn ApGen")="U"
  CLEAR WINDOWS
  CLEAR ALL
  CLOSE ALL
  CLOSE PROCEDURE
  gn_ApGen=1
ELSE
  gn_ApGen=gn_ApGen+1
  IF gn_ApGen > 4
    Do Pause WITH "Maximum level of Application nesting exceeded."
  RETURN
ENDIF
PRIVATE gn_oldsize
gn_oldsize=gn_scrsize
PRIVATE gc_bell, gc_carry, gc_clock, gc_century, gc_confirm, gc_deli,;
gc_safety, gc_status, gc_score, gc_talk, gl_leave, gc_prognum,;
gc_quit, gc_color, gc_display, gl_color, gl_batch, gn_scrsize

ENDIF
*-- Store some sets to variables
gc_bell =SET("BELL")
gc_carry =SET("CARRY")
gc_clock =SET("CLOCK")
gc_color =SET("ATTRIBUTE")
gc_century=SET("CENTURY")
gc_confirm=SET("CONFIRM")
gc_cursor =SET("CURSOR")
gc_deli =SET("DELIMITERS")
gc_display=SET("DISPLAY")
gc_safety =SET("SAFETY")
gc_status =SET("STATUS")
gc_score =SET("SCOREBOARD")
gc_talk =SET("TALK")
SET CONSOLE ON
ON ERROR ??
SET DISPLAY TO MONO
SET DISPLAY TO COLOR
SET DISPLAY TO EGA25
ON ERROR

SET BELL ON
SET CARRY OFF
SET CENTURY OFF
SET CLOCK OFF
SET CONFIRM OFF
SET DEFAULT TO F
SET DELIMITERS TO ""
SET DELIMITERS OFF
SET DEVICE TO SCREEN
SET ESCAPE OFF
SET EXCLUSIVE OFF
SET LOCK ON
SET MESSAGE TO ""
SET PATH TO \dbtest
SET PRINT OFF
SET REPROCESS TO 4
SET SAFETY OFF
SET TALK OFF

*-- Initialize global variables
gl_batch=.f.      && is a batch operation in progress
gl_color= ISCOLOR() .AND. SET("DISPLAY") <> "CGAMONO"
gn_error=0       && 0 if no error, otherwise an error occurred
gn_scrsize=21    && number of lines on screen
gn_send=0        && return value from popup of position menus
gn_trace=1       && sets trace level, however you need to change template
gc_brdr='1'      && border to use when drawing boxes
gc_dev='CON'      && Device to use for printing - See Proc. PrintSet
gl_leave=.f.     && leave the application
gc_prognum=' '    && internal program counter to handle nested menus
gc_quit=' '       && memvar for return to caller
gc_scope=' '      && scope, for and while of position at runtime
listval='NO_FIELD' && Pick List value

*-- remove asterisk to turn clock on

```

```

* SET CLOCK TO

*-- Blank the screen
SET COLOR TO
CLEAR
SET SCOREBOARD OFF
SET STATUS OFF

*-- Define menus
DO MPDEF                                && Menu Process DEFINition

*-- Execute main menu
DO WHILE .NOT. gl_leave
DO R M BAT WITH " 00"
IF gc_quit = 'Q'
EXIT
ENDIF
gl_leave = _NodShake( " ; Do you want to leave this application? ", ;
13, 18, 2, 44, .T. )
ENDDO

*-- Reset environment
DEACTIVATE WINDOW FullScr
?? Color(gc_color)
gn_ApGen=gn_ApGen-1
SET BELL &gc_bell.
SET CARRY &gc_carry.
SET CLOCK &gc_clock.
SET CENTURY &gc_century.
SET CONFIRM &gc_confirm.
SET CURSOR &gc_cursor.
SET DELIMITERS &gc_deli.
SET DISPLAY TO &gc_display.
SET STATUS &gc_status.
SET SAFETY &gc_safety.
SET SCOREBOARD &gc_score.
SET TALK &gc_talk.

IF gn_ApGen < 1
ON KEY LABEL F1
CLEAR WINDOWS
CLEAR ALL
CLOSE ALL
CLOSE PROCEDURE
SET ESCAPE ON
SET MESSAGE TO ""
CLEAR
ELSE
DEFINE WINDOW FullScr FROM 0,0 TO gn_oldsiz+3,79 NONE
DEFINE WINDOW Savescr FROM 0,0 TO gn_oldsiz,79 NONE
DEFINE WINDOW Helpscr FROM 0,0 TO gn_oldsiz,79 NONE
ACTIVATE WINDOW FullScr
ENDIF

RETURN
*-- EOP: F:R MOUTH

*-----
* Description..: Procedure files for generated menu system.
* The programs that follow are common to main routines
* The last procedure is the Menu Process DEFINition
*-----

PROCEDURE Lockit
PARAMETER pc_ltype
*-----
* Lock the current file or record based on the value of <pc_ltype>.
*-----
IF NETWORK()
gn_error=0
ON ERROR DO Multerr
IF pc_ltype = "1"
1) Tock=FLOCK()
ENDIF
IF pc_ltype = "2"
1) Tock=RLOCK()
ENDIF
ON ERROR
ENDIF
RETURN
*-- EOP: Lockit WITH pc_ltype

PROCEDURE Info_Box
PARAMETERS pc_say
*-----
* Display the message <pc_say> inside of boxes.

```

```

*-----
? pc say
? REPLICATE("-",LEN(pc_say))
?
RETURN
*--EOP: Info_Box WITH pc_say

PROCEDURE ShowPick
*-----
* Show pick list values.
*-----
PRIVATE ln_ikey, x1, x2

listval=PROMPT()
IF LEFT(entryflg,1)="B"
  lc_file=POPUP()
  DO &lc_file. WITH "A"
  RETURN
ENDIF
IF TYPE("lc_window")="U"
  ACTIVATE WINDOW ShowPick
ELSE
  ACTIVATE WINDOW &lc_window.
ENDIF

STORE 0 TO ln_ikey,x1,x2
ln_ikey=LASTKEY()
IF ln_ikey=13
  x1=AT(TRIM(listval)+',',lc_fldlst)
  IF x1 = 0
    lc_fldlst=lc_fldlst+TRIM(listval)+','
  ELSE
    x2=AT(', ',SUBSTR(lc_fldlst,x1))
    lc_fldlst=STUFF(lc_fldlst,x1,x2,'')
  ENDIF
  CLEAR
  ? lc_fldlst
ENDIF
ACTIVATE SCREEN

RETURN
*--EOP: ShowPick

PROCEDURE Cleanup
*-----
* Test whether report option was selected
*-----

DO CASE

CASE gc_dev='CON'
  DO Wait4Key

CASE gc_dev='PRN'
  SET PRINT OFF
  SET PRINTER TO

CASE gc_dev='TXT'
  CLOSE ALTERNATE

ENDCASE
gc_dev='CON'

RETURN
*-- EOP: Cleanup

PROCEDURE Pause
PARAMETER pc_msg
*-----
* Procedure to display a message or errors in a window
* Parameters : pc_msg = message line
*-----
PRIVATE lc_msg

IF TYPE("lc_message")="U"
  gn_error=ERROR()
ENDIF
lc_msg = pc_msg
lc_option='0'

ACTIVATE WINDOW Pause
IF gn_error > 0
  IF TYPE("lc_message")="U"
    @ 0,1 SAY [An error has occurred !! - Error message: ]+MESSAGE()
  ELSE
    @ 0,1 SAY [Error # ]+lc_message
  ENDIF
ENDIF

```

```

    ENDIF
ENDIF

@ 1,1 SAY lc_msg
WAIT " Press any key to continue..."
DEACTIVATE WINDOW Pause

RETURN
*-- EOP: Pause WITH pc_msg

PROCEDURE Multerr
*-----
* Errors will be handled in one of two ways:
*
* 1) Display message and return to menu
* 2) Dialog box for options Try again and Return to menu
*-----
PRIVATE lc_opt, lc_erno

gn_error=ERROR()                && Set the global error variable
lc_erno=STR(ERROR(),3)+'.'      && Contains error number to test
lc_opt='T'                       && Option variable

*-- Dialog box for options Try again and Return to menu.
IF lc_erno $ "108,109,128,129,"
ACTIVATE WINDOW Pause
@ 0,2 SAY lc_erno+" "+MESSAGE()
@ 2,22 SAY "T = Try again, R = Return to menu." ;
GET lc_opt PICTURE "I" VALID lc_opt $ "TR"

READ
DEACTIVATE WINDOW Pause
IF lc_opt = "R"
RETURN
ENDIF
ENDIF

*-- Display message and return to menu.
IF .NOT. lc_erno $ "108,109,128,129,"
DO PAUSE WITH ERROR()
RETURN
ENDIF

gn_error=0                       && Reset global variable
RETRY                             && Try the command again

RETURN
*-- EOP: Multerr

PROCEDURE Trace
PARAMETERS pc_msg, pn_lvl
*-----
* DESCRIPTION
* Trace procedure to let programmer know what module
* is about to execute and what module has executed.
*
* PARAMETERS
* pc_msg = message line
* pn_lvl = trace level
*-----
PRIVATE lc_msg, ln_lvl, lc_trp

lc_msg = pc_msg
ln_lvl = pn_lvl
lc_trp = ' '
IF gn_trace < ln_lvl                && Check for trace off
RETURN
ENDIF

DEFINE WINDOW trace FROM 5,0 TO 16,79 DOUBLE
ACTIVATE WINDOW trace

DO WHILE lc_trp <> 'Q'
CLEAR
@ 2,40-LEN(lc_msg)/2 SAY lc_msg
@ 4,05 SAY 'S - Set trace level, D - Display status, M - display Memory'
@ 5,05 SAY 'P - Turn printer on, Q - to Quit'
lc_trp = 'Q'
@ 5,38 GET lc_trp PICTURE "@M Q,S,D,M,P"
READ

DO CASE
CASE lc_trp = 'S'
@ 2,01 CLEAR
@ 2,33 SAY 'Set trace level'

```

```

@ 4,05 SAY 'Enter trace level to change to:' GET gn_trace PICTURE '#'
@ 5,05 SAY [ '0' turns trace off]
READ

IF gn_trace=0
@ 2,01 CLEAR
@ 3,05 SAY 'Trace is now turned off..To reactivate Trace - Press [F3]'
@ 4,05 say 'Press any key to continue...'
WAIT ''
ON KEY LABEL F3 gn_trace = 1
ENDIF

CASE lc_trp = 'D'
DISPLAY STATUS
WAIT

CASE lc_trp = 'M'
DISPLAY MEMORY
WAIT

CASE lc_trp = 'P'
SET PRINT ON

ENDCASE
ENDDO

SET PRINT OFF
RELEASE WINDOW trace
@ 24,79 SAY " "

RETURN
*-- EOP: Trace WITH pc_msg, pn_lvl

PROCEDURE PrintSet
*-----
* Procedure for toggling printer output
*-----
PRIVATE lc_window

gc_dev='CON'
lc_choice=' '
gn_pkey=0
gn_send=3

DEFINE WINDOW printemp FROM 08,25 TO 17,56

DEFINE POPUP SavePrin FROM 10,40
DEFINE BAR 1 OF SavePrin PROMPT " Send output to ..." SKIP
DEFINE BAR 2 OF SavePrin PROMPT REPLICATE(CHR(196),24) SKIP
DEFINE BAR 3 OF SavePrin PROMPT " CON: Console" MESSAGE "Send output to Screen"
DEFINE BAR 4 OF SavePrin PROMPT " LPT1: Parallel port 1 " MESSAGE "Send output to LPT1:"
DEFINE BAR 5 OF SavePrin PROMPT " LPT2: Parallel port 2" MESSAGE "Send output to LPT2:"
DEFINE BAR 6 OF SavePrin PROMPT " COM1: Serial port 1" MESSAGE "Send output to COM1:"
DEFINE BAR 7 OF SavePrin PROMPT " FILE = REPORT.TXT" MESSAGE "Send output to File Report.tx
t"
ON SELECTION POPUP SavePrin DEACTIVATE POPUP

lc_window = WINDOW()
IF .NOT. ISBLANK( lc_window )
    && Capture current window name
    && If window was active
    ACTIVATE SCREEN
    && Activate screen for correct popup posit
ENDIF

ACTIVATE POPUP SavePrin
gn_send = BAR()
RELEASE POPUP SavePrin

IF .NOT. ISBLANK( lc_window )
    && If window was active before
    && Reactivate it
ENDIF

IF gn_send <> 0
    && If user made a popup selection

    IF gn_send = 7
        && Output to file
        gc_dev = 'TXT'
        SET ALTERNATE TO REPORT.TXT
        SET ALTERNATE ON
    ELSE
        IF gn_send <> 3
            && Output not to the screen
            gc_dev = 'PRN'
            ON ERROR DO prntrtry
            DO CASE
                CASE gn_send = 4
                    SET PRINTER TO LPT1
                CASE gn_send = 5
                    SET PRINTER TO LPT2
            END CASE
        END IF
    END IF
END IF

```

```

        CASE gn_send = 6
            SET PRINTER TO COM1
        ENDCASE
        SET PRINT ON
        ON ERROR
    ENDF
ENDIF

ELSE
    gn_pkey = 27
    ENDIF
    && Signal escape pressed to caller

RELEASE WINDOW printemp

RETURN
*-- EOP: PrintSet

PROCEDURE PrntRtry
*-----
* On error routine for handling printer errors.
*-----
PRIVATE ll_escape
ll_escape = SET("ESCAPE") = "ON"

IF .NOT. PRINTSTATUS()
    && If printer not ready

    IF ll_escape
        && If Escape on, set it off
        SET ESCAPE OFF
    ENDIF

    ACTIVATE WINDOW printemp
    @ 1,0 SAY "Please ready your printer or"
    @ 2,0 SAY "      press ESC to cancel"

    *-- Loop until printer is ready or user presses escape
    gn_pkey = 0
    DO WHILE (.NOT. PRINTSTATUS()) .AND. gn_pkey <> 27
        gn_pkey = INKEY()
    ENDDO

    DEACTIVATE WINDOW printemp
    IF ll_escape
        SET ESCAPE ON
    ENDIF

    IF gn_pkey <> 27
        && If user wants to retry
        RETRY
        && Retry the print command
    ENDIF

ENDIF

RETURN
*-- EOP: PrntRtry

PROCEDURE Position
*-----
* Record positioning (Ask at runtime)
*-----

IF LEN(DBF()) = 0
    && Make sure a DBF is open
    DO Pause WITH "Database not in use. "
    RETURN
ENDIF

IF EOF()
    && If end of file, go to top
    GOTO TOP
ENDIF

ll_delim = SET("DELIMITERS") = "ON"
SET DELIMITERS OFF
ll_space = SET("SPACE") = "ON"
SET SPACE ON

ln_type = 0
ln_rkey = READKEY()
ln_rec = RECHO()
ln_num = 0
ll_logic = .T.
ld_date = DATE()
lc_option = '0'
lc_inl = SPACE(100)
&& Sublevel selection
&& Test for ESC or Return
&& DBF record number
&& For input of a number
&& For input of a logical
&& For input of a date
&& Main option ie. Seek, Goto and Locate
&& For input of a character

*-- Scope ie. ALL, REST, NEXT <n>
STORE SPACE(10) TO lc_scp

*-- 1 = Character SEEK, 2 = For clause, 3 = While clause

```

```

STORE SPACE(40) TO lc_ln2, lc_ln3
lc_temp = ""
gc_scope = ""
@ 0,00 SAY "Index order: "+IIF(" "=ORDER(),"Database is in natural order",ORDER())
@ 1,00 SAY "Listed below are the first 16 fields."
lc_temp=REPLICATE(CHR(196),19)
@ 2,0 SAY CHR(218)+lc_temp+CHR(194)+lc_temp+CHR(194)+lc_temp+CHR(194)+lc_temp
ln_num=240

*-- Build a list of fields in the current DBF
DO WHILE ln_num < 560
  lc_temp = FIELD( (ln_num-240)/20 +1)
  @ (ln_num/80),MOD(ln_num,80) SAY CHR(179)+;
  lc_temp+SPACE(11-LEN(lc_temp))+;
  SUBSTR("= Char = Date = Logic = Num = Float = Memo
  ";
  AT(TYPE(lc_temp),"CDLNFMU")*8-7,8)
  ln_num=ln_num+20
ENDDO
ln_num=1

DEFINE POPUP Posit1 FROM 8,30
DEFINE BAR 1 OF Posit1 PROMPT " Position by " SKIP
DEFINE BAR 2 OF Posit1 PROMPT REPLICATE(CHR(196),21) SKIP
DEFINE BAR 3 OF Posit1 PROMPT " SEEK Record" MESSAGE "Search on index key" SKIP FOR " "=ORDE
R()
DEFINE BAR 4 OF Posit1 PROMPT " GOTO Record" MESSAGE "Position to specific record"
DEFINE BAR 5 OF Posit1 PROMPT " LOCATE Record " MESSAGE "Locate record for condition"
DEFINE BAR 6 OF Posit1 PROMPT REPLICATE(CHR(196),21) SKIP
DEFINE BAR 7 OF Posit1 PROMPT " Change index order " MESSAGE "Reassign current index order
"
DEFINE BAR 8 OF Posit1 PROMPT " Done positioning " ;
MESSAGE "Press return here, when ready " +;
"(ESC cancels selections, except index order)"
ON SELECTION POPUP Posit1 DEACTIVATE POPUP

SET CONFIRM ON
gn_send = 0
DO WHILE gn_send = 0

  ACTIVATE POPUP Posit1
  gn_send = BAR()
  lc_option = ltrim(str(gn_send))      && Convert bar# to string

  IF gn_send = 0
    gc_scope=""
    GOTO ln_rec
    EXIT
  ENDF

  gn_send=0

  DO CASE

    CASE lc_option='3'                && Seek record

      IF LEN(NDX(1))=0 .AND. LEN(MDX(1))=0
        DO Pause WITH "Can't use this option - No index files are open."
        LOOP
      ENDF
      lc ln1=SPACE(100)
      DEFINE WINDOW Posit2 FROM 8,14 TO 17,67 DOUBLE
      ACTIVATE WINDOW Posit2
      keyexpr=""
      lc type = Get Type( ORDER() )
      @ 1,1 SAY "Key expression:"
      @ 2,1
      ?? keyexpr FUNCTION "V50"
      SET CONFIRM ON
      @ 5,1 SAY "Enter the key expression to search for:"
      DO CASE
        CASE lc type = "C"
          @ 6,1 GET lc ln1 FUNCTION "S50"
        CASE lc type = "D"
          @ 6,1 GET ld_date FUNCTION "D"
        CASE lc type = "L"
          @ 6,1 GET ll_logic
        CASE lc type = "N" .OR. lc type = "F"
          @ 6,1 GET ln_num PICTURE "#####.#####"
      ENDCASE
      SET CURSOR ON
      READ
      SET CURSOR OFF
      SET CONFIRM OFF
      RELEASE WINDOWS Posit2

```



```

IF .NOT. (READKEY() = 12 .OR. READKEY() = 268)
DO CASE
CASE lc_type = "C"
SEEK TRIM(lc_ln1)
CASE lc_type = "D"
SEEK Td_date
CASE lc_type = "L"
SEEK Tl_logic
CASE lc_type = "N" .OR. lc_type = "F"
SEEK Tn_num
ENDCASE
ELSE
GOTO TOP
LOOP
ENDIF

CASE lc_option='4' && Go to

ln_type=1
DEFINE POPUP Posit2 FROM 8,30
DEFINE BAR 1 OF Posit2 PROMPT " GOTO:" SKIP
DEFINE BAR 2 OF Posit2 PROMPT REPLICATE(CHR(196),10) SKIP
DEFINE BAR 3 OF Posit2 PROMPT " TOP" MESSAGE "GOTO Top of File"
DEFINE BAR 4 OF Posit2 PROMPT " BOTTOM" MESSAGE "GOTO Bottom of File"
DEFINE BAR 5 OF Posit2 PROMPT " Record # " MESSAGE "GOTO A Specific Record"
ON SELECTION POPUP Posit2 DEACTIVATE POPUP

ACTIVATE POPUP posit2
gn_send = BAR()
ln_type = gn_send
gn_send=0
IF ln_type <> 0 && If the user selected a bar

DO CASE

CASE ln_type=5 && GOTO A Specific Record

DEFINE WINDOW Posit2 FROM 8,26 TO 13,50 DOUBLE
ACTIVATE WINDOW Posit2
ln_num=0
@ 3,1 SAY "Max. Record # = "+LTRIM(STR(RECCOUNT()))
@ 1,1 SAY "Record to GOTO" GET ln_num PICT "#####" RANGE 1,RECCOUNT()
SET CURSOR ON
READ
SET CURSOR OFF
IF .NOT. (READKEY() = 12 .OR. READKEY() = 268)
GOTO ln_num
ENDIF
RELEASE WINDOWS Posit2

CASE ln_type=4 && GOTO Bottom of file
GOTO BOTTOM

CASE ln_type=3 && GOTO Top of file
GOTO TOP

ENDCASE

ENDIF

CASE lc_option='5' && Locate

DEFINE WINDOW Posit2 FROM 8,16 TO 14,66 DOUBLE
ACTIVATE WINDOW Posit2
@ 1,19 SAY "ie. ALL, NEXT <n>, and REST"
@ 1,01 SAY "Scope:" GET lc_scp
@ 2,01 SAY "For: " GET lc_ln2
@ 3,01 SAY "While:" GET lc_ln3
SET CURSOR ON
READ
SET CURSOR OFF

IF .NOT. (READKEY() = 12 .OR. READKEY() = 268)
lc_temp=TRIM(lc_scp)
lc_temp=lc_temp + IIF(LEN(TRIM(lc_ln2)) > 0," FOR "+TRIM(lc_ln2),"")
lc_temp=lc_temp + IIF(LEN(TRIM(lc_ln3)) > 0," WHILE "+TRIM(lc_ln3),"")
IF .NOT. ISBLANK(lc_temp)
LOCATE &lc_temp.
IF .NOT. EOF()
gc_scope=lc_temp
ENDIF
ELSE
DO Pause WITH "All fields were blank."
ENDIF
ENDIF
RELEASE WINDOW Posit2

```

```

CASE lc_option='7'                                && Change index order
  DO Showtag
  IF LASTKEY() = 27
    LOOP
  ENDIF
  GOTO TOP
  In rec=RECNO()
  @ 0,00
  @ 0,00 SAY "Index order: "+IIF(" "=ORDER(),"Database is in natural order",ORDER())

CASE lc_option='8'                                && User wants to exit
  gn_send=1                                        && Signal end of loop

ENDCASE

IF EOF()
  DO Pause WITH "Record not found."
  GOTO In rec
  gn_send=0
ENDIF
ENDDO
SET CURSOR ON
IF !! deli
  SET DELIMITERS ON
ENDIF
IF .NOT. !! space
  SET SPACE OFF
ENDIF
SET CONFIRM OFF
IF ISBLANK( lc_scp )
  gc_scope="REST "+gc_scope
ENDIF

RETURN
*--EOP: Position

PROCEDURE Showtag
*-----*
* Display a list of tags for the current file
*-----*
PRIVATE cnt, idxexp, length

cnt      = 1                                && Count of index tags
idxexp   = ""                               && Index tag expression
length   = 12                               && Minimum tag width

DEFINE POPUP Showtag FROM 8,10
DEFINE BAR 1 OF Showtag PROMPT " Index Order" SKIP
DEFINE BAR 3 OF Showtag PROMPT " Natural Order "

*-- For each TAG define a menu bar
DO WHILE .NOT. ISBLANK( TAG( cnt ) )
  scnt = LTRIM(STR(cnt+4))                  && Add four to bar number
  string = SUBSTR(TAG(cnt)+SPACE(10),1,11)+CHR(179)+" "+LEFT(KEY(cnt),47)
  IF length < LEN(string)                  && Adjust popup width as needed
    length = LEN(string)
  ENDIF

  DEFINE BAR &scnt. OF Showtag PROMPT string
  cnt = cnt + 1
ENDDO

DEFINE BAR 2 OF Showtag PROMPT REPLICATE(CHR(205),length+1) SKIP
DEFINE BAR 4 OF Showtag ;
  PROMPT REPLICATE(CHR(196),11)+CHR(194)+REPLICATE(CHR(196),length-11) SKIP
ON SELECTION POPUP Showtag DEACTIVATE POPUP

IF cnt > 1                                    && If the DBF has tags
  xx=4                                        && Right arrow
  DO WHILE xx = 4 .OR. xx = 19              && Ignore escape from arrow keys
    ACTIVATE POPUP Showtag
    idxexp = PROMPT()
    xx=LASTKEY()
  ENDDO

  IF BAR() <> 0                              && If user selected a tag
    IF LTRIM(RTRIM(idxexp))="Natural Order"
      SET ORDER TO
    ELSE
      lc_ord = TRIM(LEFT(idxexp,10))
      lc_exp = TRIM(SUBSTR(idxexp,14))
      SET ORDER TO &lc_ord.
    ENDIF
  ENDIF
ENDIF

```

```

ENDIF
RELEASE POPUP Showtag

RETURN
*--EOP: Showtag

FUNCTION Get Type
PARAMETER tagname
-----
* Determine the type of the tag name.
* Also will set the value of <keyexpr> for the calling routine.
-----
PRIVATE ll_exact, ll_talk, keytype, lc_temp, offset, ;
        ln_error

IF SET( "TALK" ) = "ON"
    SET TALK OFF
    ll_talk = .T.
ELSE
    ll_talk = .F.
ENDIF
ll_exact = SET("EXACT") = "ON"
SET EXACT ON

keytype = "U"           && Assume undefined tag
lc_temp = "U"           && Assume error occured with undefined key
offset = 1              && Counter for tags in DBF
ndxflag = .T.
ln_error = 0

DO WHILE .NOT. ISBLANK( TAG( offset ) )

    IF TAG(offset) = tagname
        lc_temp = TYPE( KEY( offset ) )
        ln_error = 0
        ON ERROR ln_error = ERROR()

        DO CASE
            CASE lc_temp="C"
                SEEK "A"
            CASE lc_temp="D"
                SEEK DATE()
            CASE lc_temp="L"
                SEEK .T.
            CASE lc_temp="N" .OR. lc_temp="F"
                SEEK I
        ENDCASE

        ON ERROR
        IF ln_error = 0
            keytype = lc_temp
            keyexpr = KEY(offset)
            EXIT
        ENDIF
    ENDIF

    offset = offset + 1

ENDDO

IF .NOT. ll_exact
    SET EXACT OFF
ENDIF
IF ll_talk
    SET TALK ON
ENDIF

RETURN( keytype )
*--EOP: Get_Type( Tagname )

PROCEDURE BefAct
-----
* Save the screen before executing a menu option.
-----
SAVE SCREEN TO Browscr&lc_ApGen.
DEACTIVATE WINDOW Fullscr
SET SCOREBOARD ON
RETURN
*--EOP: BefAct

PROCEDURE AftAct
-----
* Restore the screen after executing a menu option.
-----
CLEAR

```

```

SET SCOREBOARD OFF
ACTIVATE WINDOW Fullscr
RESTORE SCREEN FROM Browscr&lc_ApGen.
RELEASE SCREEN Browscr&lc_ApGen.
RETURN
*--EOP: AftAct

```

```

PROCEDURE Postnhlp

```

```

*-----
* Display help screens for generic menus.
*-----

```

```

DEFINE WINDOW Temphelp FROM 3,12 TO 19,67
ACTIVATE WINDOW Temphelp
DO CASE
CASE "SEEK" $ PROMPT()
*-- HELP SEEK
? " SEEK <exp>"
?
? " Evaluates a specified expression and attempts to"
? " find its value in the master index of the database"
? " file. Returns a logical true (.T.) if the index"
? " key is found, and a logical false (.F.) if it is"
? " not found."
?
? " Ex: SEEK CTOD('11/03/87') - converts the"
? " expression from character to date and"
? " then searches for the value in the index"
?
CASE LEFT(LTRIM(PROMPT()),4) $ "GOTO TOP BOTT Reco"
*-- HELP GOTO
? " GO/GOTO BOTTOM/TOP [IN <alias>]"
? " or"
? " GO/GOTO [RECORD] <record number> [IN <alias>]"
? " or"
? " <record number>"
?
? " Positions the record pointer to a specified record"
? " or location in the active database file."
?
? " TOP moves the pointer to the first record"
? " BOTTOM moves the pointer to the last record"
?
? " Ex: 4 - moves the record pointer to record 4"
?
CASE "LOCATE" $ PROMPT()
*-- HELP LOCATE
? " LOCATE FOR <condition> [<scope>]"
? " [WHILE <condition>]"
?
? " Searches the active database file, sequentially,"
? " for the first record that meets the specified"
? " criteria. The function FOUND() returns true (.T.)"
? " if LOCATE is successful."
?
? " Ex: LOCATE FOR Age = '25' NEXT 5"
? " searches for the next five records"
? " containing 25 in the Age field"
?
CASE "Change index order" $ PROMPT() .OR. POPUP() = "SHOWTAG"
?
? [ Select "Change index order" to select the master]
? " (controlling) index. You will see a list of indexes"
? " from the stand-alone indexes (.ndx), and mdx"
? " file(s) that are activated. The first option in the"
? " list, NATURAL ORDER, uses the file in its unindexed"
? " state. Press RETURN to select your choice by which"
? " to order the file."
?
ENDCASE
DO Wait4Key
DEACTIVATE WINDOW Temphelp
RELEASE WINDOW Temphelp
RETURN
*--EOP: Postnhlp

```

```

PROCEDURE Wait4Key

```

```

*-----
* Wait for a key press or mouse click.
*-----

```

```

PRIVATE ll_escape

```

```

ll_escape = SET( "ESCAPE" ) = "ON"
SET ESCAPE OFF
WAIT

```

```

IF ! escape
SET ESCAPE ON
ENDIF

RETURN
*-- EOP: Wait4Key

FUNCTION Color
PARAMETERS pc_scolor
*-----
* Format:
* COLOR( <expC> )
* <expC> = NORMAL, HIGHLIGHT, MESSAGES, TITLES, BOX, INFORMATION, FIELDS
* or a variable with all colors store in it
* Ver: dBASE 1.1
*
* The COLOR() function either returns or sets colors returned with the
* SET("attribute") setting
* If <expC> is a color string then null is returned otherwise the color
* setting is returned for one of dBASE's color options
*
* See Also: SET("attribute")
*-----
PRIVATE color_num, color_str, cnt

pc_scolor = UPPER(pc_scolor)
IF pc_scolor = "COLOR"
*-- Return standard, enhanced, border colors only
RETURN SUBSTR(SET("attr"),1, AT(" &", SET("attr")))
ENDIF

*-- Declare array to parse color options from SET("attr")
PRIVATE color
DECLARE color [8]
*-- Determine if user is restoring colors vs. saving colors
IF " &" $ pc_scolor
color_str = ","+pc_scolor+", " && Restore color attributes
ELSE
color_str = ","+SET("ATTRIBUTE")+", " && Save color attributes
ENDIF

*-- Stuff array with individual color setting
color_str = STUFF(color_str, AT(" &", color_str), 4, ",")
cnt = 1
DO WHILE cnt <= 8
color_str = SUBSTR(color_str, AT(",", color_str) + 1)
color[cnt] = SUBSTR(color_str, 1, AT(",", color_str) - 1)
cnt = cnt + 1
ENDDO

IF " &" $ pc_scolor
*-- Set color back
SET COLOR TO , &color [3]. && Border color
SET COLOR OF NORMAL TO &color [1].
SET COLOR OF HIGHLIGHT TO &color [2].
SET COLOR OF MESSAGES TO &color [4].
SET COLOR OF TITLES TO &color [5].
SET COLOR OF BOX TO &color [6].
SET COLOR OF INFORMATION TO &color [7].
SET COLOR OF FIELDS TO &color [8].
ELSE
*-- Return color string requested
DO CASE
CASE pc_scolor $ "NORMAL"
color_num = 1
CASE pc_scolor $ "HIGHLIGHT"
color_num = 2
CASE pc_scolor $ "BORDER"
color_num = 3
CASE pc_scolor $ "MESSAGES"
color_num = 4
CASE pc_scolor $ "TITLES"
color_num = 5
CASE pc_scolor $ "BOX"
color_num = 6
CASE pc_scolor $ "INFORMATION"
color_num = 7
CASE pc_scolor $ "FIELDS"
color_num = 8
ENDCASE
ENDIF
RETURN IIF(" &" $ pc_scolor, "", color_[color_num])

FUNCTION NodShake
PARAMETERS pc_mmsg, pn_up, pn_left, pn_height, pn_max, pl_dflt_no

```

```

*-----
* NAME
*   _NodShake
*
* DESCRIPTION
*   Accepts a YES/NO response from user
*
* SYNOPSIS
*   DO _NodShake WITH pc_mssg, pn_up, pn_left, pn_height, pn_max, pl_dflt_no
*
* PARAMETERS
*   pc_mssg:   dialog box message
*   pn_up:     upper coordinate of dialog box
*   pn_left:   left coordinate of dialog box
*   pn_height: height of dialog box
*   pn_max:    maximum width of a line in message
*   pl_dflt_no: flag indicating if default pad highlighted should be "NO"
*
* EXAMPLE
*   pl_set = _NodShake( pc_vermssg, 13, 25, 2, 28, .T. )
*-----

PRIVATE ll_ans, ll_console, ll_wrapset, ln_pspset

ll_console = SET( "CONSOLE" ) = "OFF"
SET CONSOLE ON
ll_wrapset = wrap
ln_pspset = _pspacing
wrap = .F.
_pspacing = 1

DEFINE WINDOW NodShake DOUBLE ;
  FROM pn_up, pn_left TO pn_up + pn_height + 4, pn_left + pn_max + 1

DEFINE MENU NodShake
DEFINE PAD Yes OF NodShake PROMPT "Yes" ;
  AT pn_height + 1, (pn_max - 12) / 2;
  MESSAGE "Select option and press ENTER, or press first letter" + ;
    " of desired option"

ON SELECTION PAD Yes OF NodShake DEACTIVATE MENU
DEFINE PAD No OF NodShake PROMPT "No" ;
  AT pn_height + 1, (pn_max - 12) / 2 + 10 ;
  MESSAGE "Select option and press ENTER, or press first letter" + ;
    " of desired option"

ON SELECTION PAD No OF NodShake DEACTIVATE MENU
ACTIVATE WINDOW NodShake
CLEAR
?
@ 0, 0
?? pc_mssg FUNCTION ";"

ON KEY LABEL Y KEYBOARD "{Alt-Y}{13}"
ON KEY LABEL N KEYBOARD "{Alt-N}{13}"

IF pl_dflt_no
  KEYBOARD "{Alt-N}"
ENDIF

ON KEY LABEL RIGHTARROW
ON KEY LABEL LEFTARROW

ACTIVATE MENU NodShake

ON KEY LABEL Y
ON KEY LABEL N

IF PAD() = "YES"
  ll_ans = .T.
ELSE
  ll_ans = .F.
ENDIF

RELEASE WINDOW NodShake
RELEASE MENU NodShake
wrap = ll_wrapset
_pspacing = ln_pspset

IF ll_console
  SET CONSOLE OFF
ENDIF

RETURN ll_ans
*-- EOF: _NodShake( pc_mssg, pn_up, pn_left, pn_height, pn_max, pl_dflt_no )

```

PROCEDURE Err\_Box  
 PARAMETERS pc\_msg

```

*-----
* NAME
*   Err_Box - Display an error box
*
* SYNOPSIS
*   DO Err_Box WITH <pc_msg>
*
* DESCRIPTION
*   Err_Box will display the <pc_msg> string in a box and prompt the
*   user to press any key to continue processing. Err_Box will display
*   the message based on the length of <pc_msg>.
*
* PARAMETERS
*   pc_msg - the error message to display in the box. If the length is
*           greater than 76, the trailing part is chopped off.
*
* EXAMPLE
*   DO Err_Box WITH "Incorrect window size"
*   Displays the message in a window as follows at row 9 on the screen:
*
*           +-----+
*           | Incorrect window size |
*           | Press any key to continue... |
*           +-----+
*
*   Note that the width of the window will increase to accommodate a longer
*   message string.
*
* LIMITATIONS
*   Truncates the message after 76 characters. Assumes an 80 character
*   wide screen. Looks best with SET CURSOR OFF.
*-----

```

```

PRIVATE lc_anykey, lc_msg, lc_msglen, lc_win, ln_press, ln_width, ll_trap,;
       ll_escape

lc_anykey = [Press any key to continue...]
ln_press  = LEN( lc_anykey )
lc_win    = WINDOW()                && Currently activated window if any
lc_msg    = LTRIM( RTRIM( pc_msg ) ) && Trimmed message
ln_msglen = LEN( lc_msg )           && Trimmed length of message
ln_width  = 0                       && Width of display area in window.
ll_escape = SET("ESCAPE") = "ON"
SET ESCAPE OFF

*-- Determine the width needed for the window:
IF ln_msglen <= ln_press
  ln_width = ln_press
ELSE
  *-- Make sure the message fits in the window:
  IF ln_msglen > 76
    lc_msg = LEFT( lc_msg, 76 )
    ln_msglen = 76
  ENDIF
  ln_width = ln_msglen
ENDIF
DEFINE WINDOW err_box FROM 9, ((76 - ln_width) + .5) / 2 ;
      TO 15, (ln_width + 83) / 2 DOUBLE
ln_width = ( ln_width + 2 )

*-- Display the message and prompt to the window and wait for a key press
ACTIVATE WINDOW err_box
@ 1, ( ln_width - ln_msglen ) / 2 SAY lc_msg
@ 3, ( ln_width - ln_press ) / 2 SAY lc_anykey
SET CONSOLE OFF                && For mouse click recognition
WAIT
SET CONSOLE ON

*-- Clean up the window display and reactivate the previous window
RELEASE WINDOW err_box
IF ISBLANK( lc_win )
  ACTIVATE SCREEN
ENDIF

IF ll_escape
  SET ESCAPE ON
ELSE
  SET ESCAPE OFF
ENDIF

RETURN

```

\*-- EOP: \_Err\_Box WITH pc\_msg

PROCEDURE MPDEF

```
*-----  
* Program.....: MPDEF  
* Author.....: K.Sasabe, National River Mouths Study  
* Date.....: 8-27-93  
* Notice.....: Japan International Cooperation Agency (JICA)  
* Generated by.: dBASE IV version 1.5  
* Description..: Defines all menus in the system for F:R_MOUTH  
*-----
```

```
IF gl color  
  SET COLOR OF NORMAL TO W+/B  
  SET COLOR OF MESSAGES TO RG+/N  
  SET COLOR OF TITLES TO W/B  
  SET COLOR OF HIGHLIGHT TO RG+/GB  
  SET COLOR OF BOX TO RG+/GB  
  SET COLOR OF INFORMATION TO B/W  
  SET COLOR OF FIELDS TO N/GB
```

```
ENDIF  
CLEAR
```

\*-- Sign-on banner

```
SET BORDER TO  
@ 7,16 TO 18,64 DOUBLE COLOR RG+/GB  
@ 9,17 SAY " Welcome to"  
@ 11,17 SAY " RIVER MOUTH INFORMATION MANEGEMENT SYSTEM"  
@ 12,17 SAY " (RMOUTH)"  
@ 14,17 SAY " DID-JICA"  
@ 16,17 SAY " dBASE IV V1.5"  
@ 8,17 FILL TO 17,63 COLOR RG+/N  
@ 24,30 SAY " Press "+CHR(17)+CHR(196)+CHR(217)+" to continue. "  
SET CONSOLE OFF  
DO Wait4Key  
SET CONSOLE ON  
CLEAR
```

\*-- Prevents clearing of menus from commands:

```
*-- SET STATUS and SET SCOREBOARD  
DEFINE WINDOW FullScr FROM 0,0 TO 24,79 NONE  
*-- Position at runtime and batch process  
DEFINE WINDOW Savescr FROM 0,0 TO 21,79 NONE
```

\*-- F1 Help

```
DEFINE WINDOW Helpscr FROM 0,0 TO 21,79 NONE
```

IF gn ApGen=1

```
*-- Pause message box  
DEFINE WINDOW Pause FROM 15,00 TO 19,79 DOUBLE
```

ENDIF

```
ACTIVATE WINDOW FullScr
```

```
@ 24,00
```

```
@ 23,00 SAY "Loading..."
```

```
SET BORDER TO DOUBLE
```

\*-- R M BAT - batch object.

```
?? ". "
```

```
SET BORDER TO DOUBLE
```

\*-- Bar

```
DEFINE MENU MAIN MESSAGE 'Position with: '+CHR(27)+CHR(26)+' - <Enter> to select choice - <F1  
> Help'
```

```
  DEFINE PAD PAD 1 OF MAIN PROMPT "Process" AT 1,2  
  ON SELECTION PAD PAD 1 OF MAIN DO ACT02  
  DEFINE PAD PAD 2 OF MAIN PROMPT "Card view" AT 1,15  
  ON SELECTION PAD PAD 2 OF MAIN DO ACT02  
  DEFINE PAD PAD 3 OF MAIN PROMPT "Browse view" AT 1,30  
  ON SELECTION PAD PAD 3 OF MAIN DO ACT02  
  DEFINE PAD PAD 4 OF MAIN PROMPT "Maintenance" AT 1,47  
  ON SELECTION PAD PAD 4 OF MAIN DO ACT02  
  DEFINE PAD PAD 5 OF MAIN PROMPT "End" AT 1,64  
  ON SELECTION PAD PAD 5 OF MAIN DO ACT02
```

```
?? ". "
```

```
SET BORDER TO DOUBLE
```

\*-- Popup

```
DEFINE POPUP PROCESS FROM 2,1 TO 7,20 ;
```

```
MESSAGE 'Position: '+CHR(27)+CHR(26)+CHR(25)+CHR(24)+' Select: '+CHR(17)+CHR(196)+CHR(217)+'  
Help: F1'
```

```
  DEFINE BAR 1 OF PROCESS PROMPT " Process Sub-menu" SKIP  
  DEFINE BAR 2 OF PROCESS PROMPT "===== " SKIP  
  DEFINE BAR 3 OF PROCESS PROMPT " Amend data"  
  DEFINE BAR 4 OF PROCESS PROMPT " Add new records"
```



ON SELECTION POPUP PROCESS DO ACT03  
?? ". "

SET BORDER TO DOUBLE

\*-- Popup  
DEFINE POPUP CARDVIEW FROM 2,14 TO 12,50 ;  
MESSAGE 'Position: '+CHR(27)+CHR(26)+CHR(25)+CHR(24)+' Select: '+CHR(17)+CHR(196)+CHR(217)+'  
Help: F1'  
DEFINE BAR 1 OF CARDVIEW PROMPT " River mouth name and state"  
DEFINE BAR 2 OF CARDVIEW PROMPT " Map/photo data"  
DEFINE BAR 3 OF CARDVIEW PROMPT " River features"  
DEFINE BAR 4 OF CARDVIEW PROMPT " River mouth features..."  
DEFINE BAR 5 OF CARDVIEW PROMPT " Navigation and fishery"  
DEFINE BAR 6 OF CARDVIEW PROMPT " Socioeconomic condition..."  
DEFINE BAR 7 OF CARDVIEW PROMPT " Present problems"  
DEFINE BAR 8 OF CARDVIEW PROMPT " Researcher's comments"  
DEFINE BAR 9 OF CARDVIEW PROMPT " Categorization and countermeasure"  
ON SELECTION POPUP CARDVIEW DO ACT04  
?? ". "

SET BORDER TO DOUBLE

\*-- Popup  
DEFINE POPUP BROWVIEW FROM 2,29 TO 10,65 ;  
MESSAGE 'Position: '+CHR(27)+CHR(26)+CHR(25)+CHR(24)+' Select: '+CHR(17)+CHR(196)+CHR(217)+'  
Help: F1'  
DEFINE BAR 1 OF BROWVIEW PROMPT " River mouth list"  
DEFINE BAR 2 OF BROWVIEW PROMPT " River features"  
DEFINE BAR 3 OF BROWVIEW PROMPT " River mouth features..."  
DEFINE BAR 4 OF BROWVIEW PROMPT " Navigation and fishery"  
DEFINE BAR 5 OF BROWVIEW PROMPT " Socioeconomic conditions..."  
DEFINE BAR 6 OF BROWVIEW PROMPT " Present problems"  
DEFINE BAR 7 OF BROWVIEW PROMPT " Categorization and countermeasure"  
ON SELECTION POPUP BROWVIEW DO ACT05  
?? ". "

SET BORDER TO DOUBLE

\*-- Popup  
DEFINE POPUP MAINTEN FROM 2,46 TO 6,73 ;  
MESSAGE 'Position: '+CHR(27)+CHR(26)+CHR(25)+CHR(24)+' Select: '+CHR(17)+CHR(196)+CHR(217)+'  
Help: F1'  
DEFINE BAR 1 OF MAINTEN PROMPT " Change password"  
DEFINE BAR 2 OF MAINTEN PROMPT " Back up file to floppy"  
DEFINE BAR 3 OF MAINTEN PROMPT " Restore file from floppy"  
ON SELECTION POPUP MAINTEN DO ACT06  
?? ". "

SET BORDER TO DOUBLE

\*-- Popup  
DEFINE POPUP END FROM 2,63 TO 7,78 ;  
MESSAGE 'Position: '+CHR(27)+CHR(26)+CHR(25)+CHR(24)+' Select: '+CHR(17)+CHR(196)+CHR(217)+'  
Help: F1'  
DEFINE BAR 1 OF END PROMPT " End Sub-menu" SKIP  
DEFINE BAR 2 OF END PROMPT " =====" SKIP  
DEFINE BAR 3 OF END PROMPT " End session"  
DEFINE BAR 4 OF END PROMPT " Quit to DOS"  
ON SELECTION POPUP END DO ACT07  
?? ". "

SET BORDER TO DOUBLE

\*-- Popup  
DEFINE POPUP AMEND FROM 6,2 TO 18,13 ;  
MESSAGE 'Position: '+CHR(27)+CHR(26)+CHR(25)+CHR(24)+' Select: '+CHR(17)+CHR(196)+CHR(217)+'  
Help: F1'  
DEFINE BAR 1 OF AMEND PROMPT " RESEAR 1"  
DEFINE BAR 2 OF AMEND PROMPT " RES 1 F1"  
DEFINE BAR 3 OF AMEND PROMPT " RES 1 F2"  
DEFINE BAR 4 OF AMEND PROMPT " RES 1 F3"  
DEFINE BAR 5 OF AMEND PROMPT " RES 1 F4"  
DEFINE BAR 6 OF AMEND PROMPT " RES 1 F5"  
DEFINE BAR 7 OF AMEND PROMPT " RES 1 F6"  
DEFINE BAR 8 OF AMEND PROMPT " RESEAR 2"  
DEFINE BAR 9 OF AMEND PROMPT " RESEAR 3"  
DEFINE BAR 10 OF AMEND PROMPT " ADD INFO"  
DEFINE BAR 11 OF AMEND PROMPT " MAPPHOTO"  
ON SELECTION POPUP AMEND DO ACT08  
?? ". "

SET BORDER TO DOUBLE

\*-- Popup  
DEFINE POPUP ADD FROM 7,2 TO 9,13 ;

```

MESSAGE 'Position: '+CHR(27)+CHR(26)+CHR(25)+CHR(24)+' Select: '+CHR(17)+CHR(196)+CHR(217)+'
Help: F1'
  DEFINE BAR 1 OF ADD PROMPT " MAPPHOTO"
ON SELECTION POPUP ADD DO ACT09
?? ". "

SET BORDER TO DOUBLE

*-- Popup
DEFINE POPUP C MOUTH FROM 7,15 TO 11,36 ;
MESSAGE 'Position: '+CHR(27)+CHR(26)+CHR(25)+CHR(24)+' Select: '+CHR(17)+CHR(196)+CHR(217)+'
Help: F1'
  DEFINE BAR 1 OF C MOUTH PROMPT " Geomorphology"
  DEFINE BAR 2 OF C MOUTH PROMPT " Oceanographic data"
  DEFINE BAR 3 OF C MOUTH PROMPT " Structures..."
ON SELECTION POPUP C MOUTH DO ACT10
?? ". "

SET BORDER TO DOUBLE

*-- Popup
DEFINE POPUP C SOCIO FROM 9,15 TO 15,54 ;
MESSAGE 'Position: '+CHR(27)+CHR(26)+CHR(25)+CHR(24)+' Select: '+CHR(17)+CHR(196)+CHR(217)+'
Help: F1'
  DEFINE BAR 1 OF C SOCIO PROMPT " Water use"
  DEFINE BAR 2 OF C SOCIO PROMPT " Land use, vegetation & urban centers"
  DEFINE BAR 3 OF C SOCIO PROMPT " Development plans"
  DEFINE BAR 4 OF C SOCIO PROMPT " Fauna/flora"
  DEFINE BAR 5 OF C SOCIO PROMPT " Activities and others"
ON SELECTION POPUP C SOCIO DO ACT011
?? ". "

SET BORDER TO DOUBLE

*-- Popup
DEFINE POPUP B MOUTH FROM 6,30 TO 11,53 ;
MESSAGE 'Position: '+CHR(27)+CHR(26)+CHR(25)+CHR(24)+' Select: '+CHR(17)+CHR(196)+CHR(217)+'
Help: F1'
  DEFINE BAR 1 OF B MOUTH PROMPT " Geomorphology"
  DEFINE BAR 2 OF B MOUTH PROMPT " Classification"
  DEFINE BAR 3 OF B MOUTH PROMPT " Oceanographical data"
  DEFINE BAR 4 OF B MOUTH PROMPT " Structures"
ON SELECTION POPUP B MOUTH DO ACT012
?? ". "

SET BORDER TO DOUBLE

*-- Popup
DEFINE POPUP B SOCIO FROM 8,30 TO 11,52 ;
MESSAGE 'Position: '+CHR(27)+CHR(26)+CHR(25)+CHR(24)+' Select: '+CHR(17)+CHR(196)+CHR(217)+'
Help: F1'
  DEFINE BAR 1 OF B SOCIO PROMPT " Land Use"
  DEFINE BAR 2 OF B SOCIO PROMPT " Major Urban Centers"
ON SELECTION POPUP B SOCIO DO ACT013
?? ". "

SET BORDER TO DOUBLE

*-- Popup
DEFINE POPUP RESEAR 1 FROM 8,3 TO 12,14 ;
MESSAGE 'Position: '+CHR(27)+CHR(26)+CHR(25)+CHR(24)+' Select: '+CHR(17)+CHR(196)+CHR(217)+'
Help: F1'
  DEFINE BAR 1 OF RESEAR 1 PROMPT " Part 1/3"
  DEFINE BAR 2 OF RESEAR 1 PROMPT " Part 2/3"
  DEFINE BAR 3 OF RESEAR 1 PROMPT " Part 3/3"
ON SELECTION POPUP RESEAR_1 DO ACT014
?? ". "

SET BORDER TO DOUBLE

*-- Popup
DEFINE POPUP RESEAR 2 FROM 15,3 TO 19,14 ;
MESSAGE 'Position: '+CHR(27)+CHR(26)+CHR(25)+CHR(24)+' Select: '+CHR(17)+CHR(196)+CHR(217)+'
Help: F1'
  DEFINE BAR 1 OF RESEAR 2 PROMPT " Part 1/3"
  DEFINE BAR 2 OF RESEAR 2 PROMPT " Part 2/3"
  DEFINE BAR 3 OF RESEAR 2 PROMPT " Part 3/3"
ON SELECTION POPUP RESEAR_2 DO ACT015
?? ". "

SET BORDER TO DOUBLE

*-- Popup
DEFINE POPUP RESEAR 3 FROM 16,3 TO 21,14 ;
MESSAGE 'Position: '+CHR(27)+CHR(26)+CHR(25)+CHR(24)+' Select: '+CHR(17)+CHR(196)+CHR(217)+'
Help: F1'

```

```

DEFINE BAR 1 OF RESEAR_3 PROMPT " Part 1/4"
DEFINE BAR 2 OF RESEAR_3 PROMPT " Part 2/4"
DEFINE BAR 3 OF RESEAR_3 PROMPT " Part 3/4"
DEFINE BAR 4 OF RESEAR_3 PROMPT " Part 4/4"
ON SELECTION POPUP RESEAR_3 DO ACT016
?? ". "

```

```

SET BORDER TO DOUBLE

```

```

*-- Popup
DEFINE POPUP C M STR FROM 11,16 TO 14,38 ;
MESSAGE 'Position: '+CHR(27)+CHR(26)+CHR(25)+CHR(24)+' Select: '+CHR(17)+CHR(196)+CHR(217)+'
Help: F1'
DEFINE BAR 1 OF C M STR PROMPT " General information"
DEFINE BAR 2 OF C M STR PROMPT " Details"
ON SELECTION POPUP C M STR DO ACT017
?? ". "

```

```

RETURN

```

```

*-- EOP: MPDEF.PRG

```

```

PROCEDURE 1HELP1

```

```

*-----
* Handle F1 - Help during program execution
*-----

```

```

PRIVATE lc_popmenu, ll_errbox, ll_status

```

```

ON KEY LABEL F1
lc_popmenu=IIF( "" = POPUP(), MENU(), POPUP() )
ll_status = SET( "STATUS" ) = "ON"
IF ll_status
SAVE SCREEN TO ls_status
SET STATUS OFF
RESTORE SCREEN FROM ls_status
ELSE
ACTIVATE WINDOW Helpscr
ENDIF
SET ESCAPE OFF
ACTIVATE SCREEN
@ 0,0 CLEAR TO 21,79
@ 1,0 TO 21,79 COLOR RG+/GB
@ 24,00
@ 0,0 SAY ""
ll_errbox = .F.
DO CASE

```

```

*-- help for menu R M BAT
CASE "R M BAT" = lc_popmenu
DO Err_Box WITH "No Help defined."
ll_errbox = .T.

```

```

*-- help for menu MAIN
CASE "MAIN" = lc_popmenu
DO Err_Box WITH "No Help defined."
ll_errbox = .T.

```

```

*-- help for menu PROCESS
CASE "PROCESS" = lc_popmenu
DO Err_Box WITH "No Help defined."
ll_errbox = .T.

```

```

*-- help for menu CARDVIEW
CASE "CARDVIEW" = lc_popmenu
DO Err_Box WITH "No Help defined."
ll_errbox = .T.

```

```

*-- help for menu BROWVIEW
CASE "BROWVIEW" = lc_popmenu
DO Err_Box WITH "No Help defined."
ll_errbox = .T.

```

```

*-- help for menu MAINT
CASE "MAINT" = lc_popmenu
DO Err_Box WITH "No Help defined."
ll_errbox = .T.

```

```

*-- help for menu END
CASE "END" = lc_popmenu
DO Err_Box WITH "No Help defined."
ll_errbox = .T.

```

```

*-- help for menu AMEND
CASE "AMEND" = lc_popmenu
DO Err_Box WITH "No Help defined."
ll_errbox = .T.

```

```

*-- help for menu ADD
CASE "ADD" = lc_popmenu
DO Err_Box WITH "No Help defined."
  ll_errbox = .T.

*-- help for menu C MOUTH
CASE "C MOUTH" = lc_popmenu
DO Err_Box WITH "No Help defined."
  ll_errbox = .T.

*-- help for menu C SOCIO
CASE "C SOCIO" = lc_popmenu
DO Err_Box WITH "No Help defined."
  ll_errbox = .T.

*-- help for menu B MOUTH
CASE "B MOUTH" = lc_popmenu
DO Err_Box WITH "No Help defined."
  ll_errbox = .T.

*-- help for menu B SOCIO
CASE "B SOCIO" = lc_popmenu
DO Err_Box WITH "No Help defined."
  ll_errbox = .T.

*-- help for menu RESEAR_1
CASE "RESEAR_1" = lc_popmenu
DO Err_Box WITH "No Help defined."
  ll_errbox = .T.

*-- help for menu RESEAR_2
CASE "RESEAR_2" = lc_popmenu
DO Err_Box WITH "No Help defined."
  ll_errbox = .T.

*-- help for menu RESEAR_3
CASE "RESEAR_3" = lc_popmenu
DO Err_Box WITH "No Help defined."
  ll_errbox = .T.

*-- help for menu C_M_STR
CASE "C_M_STR" = lc_popmenu
DO Err_Box WITH "No Help defined."
  ll_errbox = .T.
OTHERWISE
DO Err_Box WITH "Unknown menu name, help was never defined."
  ll_errbox = .T.
ENDCASE

IF .NOT. ll_errbox
@ 24,26 SAY "Press any key to continue..."
SET CONSOLE OFF
DO Wait4Key
SET CONSOLE ON
ENDIF

SET ESCAPE ON
@ 24,00
IF ll_status
SET STATUS ON
RESTORE SCREEN FROM ls_status
RELEASE SCREEN ls_status
ELSE
DEACTIVATE WINDOW Helpscr
ENDIF
ON KEY LABEL F1 DO 1HELP1
RETURN
*-- EOP: 1HELP1

```

```

*****
*      RMPROC.PRG
*      Procedure for River Mouth Information Management System (RIMIMAS)
*      This module is called from RIMIMAS Application
*****
*      PROCEDURE SERIALCK
*      parameter:      Return mhit= .T. or .F.
*      Function:       To check exsistence of Serial No. for renewal
*                     or deletion procedure.
*                     .T. for exsistent and .F. for non
*****

PROCEDURE SERIALCK
PARAMETERS mhit,serialno,mdo                && Logical

mserial=SPACE(3)
CLEAR TYPEAHEAD
ACTIVATE WINDOW ask1
@ 0,0 SAY "Serial No. (or '?' for name search): " ;
GET mserial PICTURE "X99"
READ

DO CASE
CASE LEN(TRIM(mserial))=0
  mhit=.T.
  mdo=.F.
CASE LEFT(mserial,1)="?"
  DO RMNAME WITH mhit,serialno
CASE SET("DELETED")="ON"
  mserialv=VAL(mserial)
  IF SEEK(mserialv) .AND. .NOT.EOF()
    serialno=mserialv
    mhit=.T.
  ELSE
    mmess1="Serial No. not found."
    mmess2=""
    DO ALARM WITH mmess1,mmess2
    mhit=.F.
  ENDIF
CASE SET("DELETED")="OFF"
  mserialv=VAL(mserial)
  IF SEEK(mserialv)
    serialno=mserialv
    mhit=.T.
  ELSE
    mmess1="Serial No. not found."
    mmess2=""
    DO ALARM WITH mmess1,mmess2
    mhit=.F.
  ENDIF
ENDCASE
DEACTIVATE WINDOW ask1

IF mhit=.T.
  ACTIVATE WINDOW name
  DO NAMESHOW
ENDIF
CLEAR TYPEAHEAD
RETURN
***** END OF PROCEDURE SERIALCK *****

*****
*      PROCEDURE ALARM
*      Parameters:    mmess1,mmess2
*      Function:      Give alarm message to WINDOW ALARM
*****

PROCEDURE ALARM
PARAMETERS mmess1,mmess2

ACTIVATE WINDOW ALARM
DO BELL1000
@ 0,0 SAY mmess1
@ 1,0 SAY mmess2
mnkey=INKEY(5)
DO BELL1000
DEACTIVATE WINDOW ALARM
RETURN

***** END OF PROCEDURE ALARM *****

*****
*      PROCEDURE RMNAME

```

```

* Parameters: mhit .T. or .F.
* Function: Display Name List corresponding to input Name and Select*
* When selected returns .T., if not returns .F.
*****
PROCEDURE RMNAME
PARAMETERS mhit,serialno

mhit=.F.
SET ORDER TO NAME && change Index Tag to NAME
GOTO TOP
mname=SPACE(15)
@ 1,0 SAY "Input Name:" GET mname
@ 1,27 SAY " ('Be' can be for 'Beruas')"
READ
mname=TRIM(mname)
mlen=LEN(mname)
IF mlen=0 && return when Name is blank
SET ORDER TO SERIAL
RETURN
ENDIF

SET EXACT OFF
SEEK mname
USE n_search IN 2 && use n_search.dbf for temporary output
nno=0
ACTIVATE WINDOW ASK2
IF .NOT. EOF()
SCAN NEXT 25 WHILE mname=LEFT(NAME,mlen)
IF DELETED()
LOOP
ENDIF
vname=NAME
recordno=RECNO()
serialno=SERIAL
mstate=STATE
SELECT 2
APPEND BLANK
REPLACE NAME WITH vname,RECORD WITH recordno,SERIAL WITH serialno,STATE WITH mstate
nno=nno+1
mname=RTRIM(NAME)
mmess=mname+" "+STATE
@ 0+nno,5 SAY mmess

IF nno>11
mmess1="Too many candidates."
mmess2="Give more information for search."
DO ALARM WITH mmess1,mmess2
CLEAR
SET CONFIRM OFF
ZAP
SET CONFIRM ON
mhit=.F.
SELECT 1
SET ORDER TO SERIAL
SET EXACT ON
RETURN
ENDIF
SELECT 1
ENDSCAN
ENDIF

SET EXACT ON
SET ORDER TO SERIAL
mchoice=0
SELECT 2
IF RECCOUNT(>0
ON SELECTION POPUP SCHOICE DO NAMEPICK WITH mchoice
ACTIVATE POPUP SCHOICE
ELSE
mmess1="No matching name."
mmess2="(Name search is case(uppercase/lowercase) sensitive)"
DO ALARM WITH mmess1,mmess2
ENDIF

IF mchoice > 0 && selected
GOTO mchoice
recordno=RECORD
serialno=SERIAL
SET CONFIRM OFF
ZAP
SET CONFIRM ON
SELECT 1
GOTO recordno
mhit=.T.
ELSE

```

```

        SET CONFIRM OFF
        ZAP
        SET CONFIRM ON
        SELECT 1
        mhit=.F.
        DEACTIVATE WINDOW ASK2
        RETURN
    ENDIF
DEACTIVATE WINDOW ASK2
RETURN

***** END OF PROCEDURE RMNAME *****

*****
*   PROCEDURE NAMEPICK
*   Parameters:  mchoice
*   Function:    Select name in Pop-up Menu.
*               Return Menu No. by mchoice.
*****

PROCEDURE NAMEPICK
PARAMETERS mchoice

mchoice=BAR()
DEACTIVATE POPUP
@ 10,08 SAY SPACE(40)
RETURN

***** END OF PROCEDURE NAMEPICK *****

*****
*   PROCEDURE NAMESHOW
*   Parameters:
*   Function:    Display selected river mouth name
*****

PROCEDURE NAMESHOW

CLEAR TYPEAHEAD
mmess=SPACE(59)
@ 0,0 SAY mmess
@ 0,0 SAY "Selected River Mouth:"+STR(SERIAL,3)+" ". "+RTRIM(NAME)+", "+RTRIM(STATE)

RETURN

***** END OF PROCEDURE NAMESHOW *****

*****
*   PROCEDURE PASSWORD
*   Parameters:  mhit
*   Function:    Check password
*****

PROCEDURE PASSWORD
PARAMETERS mhit

CLEAR TYPEAHEAD
mhit=.F.

passin=" "
ACTIVATE WINDOW pass
@ 0,0 SAY CHR(7)+"Enter Password:" GET passin
READ
@ 0,0 SAY SPACE(30)
DEACTIVATE WINDOW pass

IF LEN(TRIM(passin))=0
    RETURN
ENDIF

USE PW IN 2
SELECT 2
GOTO TOP
password=PASSWORD
SELECT 1

IF passin<>password
    mmess1="Access denied!"
    mmess2="(Password is case(uppercase/lowercase) sensitive.)"
    DO ALARM WITH mmess1,mmess2
    RETURN
ELSE
    mhit=.T.
ENDIF

```

```

RETURN
***** END OF PROCEDURE PASSWORD *****
*****
*   PROCEDURE PASSNEW
*   Parameters:
*   Function:    Check password
*
*****

PROCEDURE PASSNEW
*PARAMETERS

CLEAR TYPEAHEAD

passin=""
ACTIVATE WINDOW pass
@ 0,0 SAY CHR(7)+"Enter Old Password:" GET passin
READ
@ 0,0 SAY SPACE(30)
DEACTIVATE WINDOW pass

IF LEN(TRIM(passin))=0
  RETURN
ENDIF

USE PW IN 2
SELECT 2
GOTO TOP
password=PASSWORD
SELECT 1

IF passin<>password
  mmess1="Access denied!"
  mmess2="(Password is case(uppercase/lowercase) sensitive.)"
  DO ALARM WITH mmess1,mmess2
  RETURN
ELSE
  passin=""
  ACTIVATE WINDOW pass
  mmess=CHR(7)+"Enter New Password:"
  @ 0,0 SAY mmess GET passin
  READ
  @ 0,0 SAY SPACE(30)

  IF LEN(TRIM(passin))=0
    RETURN
  ENDIF

  DEACTIVATE WINDOW pass
  USE PW IN 2
  SELECT 2
  GOTO TOP
  REPLACE PASSWORD WITH passin
  SELECT 1
ENDIF

RETURN

***** END OF PROCEDURE PASSNEW *****
*****
*   PROCEDURE BELL1000
*   Function:    Alarm bell
*
*****

PROCEDURE BELL1000

SET BELL TO 1000,3
?CHR(7)
SET BELL TO

RETURN

***** END OF PROCEDURE BELL1000 *****
*****
*   PROCEDURE OBJRECORD
*   Function:    Check number of objective record
*
*****

PROCEDURE OBJRECORD
PARAMETERS serialno, rno

```



```

USE mapphoto IN 2
SELECT 2
SET ORDER TO SERIALKIND
GOTO TOP
serial2=STR(serialno,3,0)
SET EXACT OFF
SEEK serial2

SELECT 1
SET ORDER TO SERIAL
GOTO TOP
SEEK serialno
name3=NAME
state3=STATE

USE MAPPHOT2 IN 3
SELECT 3
REINDEX
SELECT 2

rno=0

IF .NOT. EOF()
DO WHILE SERIAL=serialno
  rno=rno+1
  serial3=SERIAL
  m p3=M P
  area3=AREA
  series3=SERIES
  scale3=SCALE
  flightlin3=FLIGHTLINE
  sheet no3=SHEET_NO
  ceb no3=CEB_NO
  date3=DATE
  time3=TIME
  source3=SOURCE

  SELECT 3
  APPEND BLANK
  REPLACE SERIAL WITH serial3, M P WITH m p3, AREA WITH area3
  REPLACE SERIES WITH series3, SCALE WITH scale3, FLIGHTLINE WITH flightlin3
  REPLACE SHEET NO WITH sheet no3, CEB NO WITH ceb no3, DATE WITH date3
  REPLACE TIME WITH time3, SOURCE WITH source3, ORDER WITH rno
  REPLACE NAME WITH name3, STATE WITH state3
  SELECT 2

  SKIP
ENDDO
ENDIF

SELECT 3
SET ORDER TO ORDER
GOTO TOP

RETURN
***** END OF PROCEDURE OBJRECORD *****

*****
* PROCEDURE P_SET
* Function: Set printer page length and pitch
*****

PROCEDURE P_SET
PARAMETERS mok,pl,pitch

mok=UPPER(mok)

DO CASE
CASE mok="Y"
  _pwait=.F.
  _plength=pl
  SET ECHO OFF
  SET TALK OFF
  SET PRINTER ON
  ?? CHR(7)+pitch
  EJECT PAGE
  SET PRINTER OFF
  SET DEVICE TO SCREEN
OTHERWISE
  SET DEVICE TO SCREEN
  _plength=23
  _pwait=.T.
ENDCASE

RETURN
***** END OF PROCEDURE P_SET *****

```

```
*****
*   PROCEDURE P_RESET
*   Function:   Reset printer page length and pitch
*****
```

```
PROCEDURE P_RESET
PARAMETERS mok
```

```
DO CASE
CASE mok="Y"
SET ECHO OFF
SET TALK OFF
SET PRINTER ON
  _p_length=60
  ?? CHR(7)+"&16D"
SET PRINTER OFF
SET DEVICE TO SCREEN
OTHERWISE
SET DEVICE TO SCREEN
  _p_length=60
  _pwait=.F.
```

```
ENDCASE
```

```
RETURN
***** END OF PROCEDURE P_RESET *****
```