

APLICAÇÃO INTERATIVA PARA MANUTENÇÃO E CONSULTA A SAS DATA
SET COM CHAVE

Lopes , Edna M.L.

APLICAÇÃO INTERATIVA PARA MANUTENÇÃO E CONSULTA A SAS DATA SET COM CHAVE*

Edna Maria Lourenção LOPES



COMISSÃO NACIONAL DE ENERGIA NUCLEAR-SP
INSTITUTO DE PESQUISAS ENERGÉTICAS E NUCLEARES
Caixa Postal 11049 - Pinheiros
05499 - São Paulo - BRASIL

RESUMO

O trabalho é uma aplicação desenvolvida em linguagem macro do SAS, usando apenas recursos do SAS básico, com o objetivo de se ter interativamente atualização e consulta a arquivos SAS com chave.

(*) Trabalho apresentado no "I Encontro de usuários SAS" - Cidade do Rio de Janeiro - Brasil, de 2 a 3 de agosto de 1989.

INTERACTIVE SYSTEM FOR UPDATING AND QUERING OF SAS DATA SETS WITH KEY*

Edna Maria Lourenção LOPES

COMISSÃO NACIONAL DE ENERGIA NUCLEAR/SP
INSTITUTO DE PESQUISAS ENERGÉTICAS E NUCLEARES
Caixa Postal 11049 - Pinheiros
05499 - São Paulo - Brasil

ABSTRACT

The paper shows how to use SAS macro language for creating an interactive system using menus for updating and quering a SAS data set with key.

(*) Paper present at "I Encontro de Usuários SAS" held in Rio de Janeiro, Brazil, Aug. 2-3, 1989.

APLICACAO PARA MANUTENCAO E CONSULTA A SAS DATA SET COM CHAVE

EDNA MARIA LOURENCAO LOPES
IPEN - CNEN / SP

I. INTRODUCAO

ESSE TRABALHO E' UMA APLICACAO DESENVOLVIDA EM SAS MACRO LANGUAGE, USANDO RECURSOS DO SAS BASE, COM O OBJETIVO DE PERMITIR A ATUALIZACAO E CONSULTA EM SAS DATA SETS, QUE CONTENHAM CAMPOS DE DADOS, COM VALOR UNICO.

A OPERACAO DO SISTEMA E' FEITA ATRAVES DE TELAS DE MENU E PERGUNTAS - RESPOSTAS.

A MANUTENCAO DO SAS DATA SET COMPREENDE A INCLUSAO, EXCLUSAO E ALTERACAO DE UMA OBSERVACAO DEFINIDA A PARTIR DE UM CAMPO CHAVE FORNECIDO PELO USUARIO. ESSE CAMPO CHAVE E' VALIDO PARA O DATA SET DURANTE A SESSAO, COM ATE' QUE OUTRO DATA SET SEJA DEFINIDO.

A CONSULTA E' FEITA SEGUNDO CRITERIO ESTABELECIDO PELO USUARIO, NA FORMA DE UMA EXPRESSAO LOGICA, INDEPENDENTE DA CHAVE, E O RESULTADO EXIBIDO EM TELA DO TIPO "FULL SCREEN".

II. OPERACAO DO SISTEMA

PARA EXEMPLIFICAR USAREMOS UM SAS DATA SET CONTENDO O CADASTRO DE FUNCIONARIOS LISTADO ABAIXO :

OBS	NFUN	NOME	CIDADE	DEPT	SEXO	SALLIQ	SALBAS
1	10052	SILVIA	SAO PAULO	918	F	48700	72900
2	10546	JOAO	SAO PAULO	914	M	43500	61800
3	10554	JOSE	SAO PAULO	918	M	66900	97200
4	10961	RONALDO	SAO PAULO	914	M	29900	39900
5	11350	PAULA	RIO	914	F	27000	39500
6	11357	CARLOS	RIO	914	M	30500	38300
7	11400	MARIA	SAO PAULO	918	F	20000	30000
8	11584	MARIA	SAO PAULO	914	F	22100	29700
9	11658	ROSA	SAO PAULO	918	F	77800	123500
10	11931	ROGERIO	RIO	914	M	101000	135000
11	12649	MARIO	RIO	918	M	55700	72800
12	2943	RICARDO	SAO PAULO	918	M	28700	40200
13	3658	EDUARDO	RIO	918	M	47900	70100
14	5781	LUCIO	SAO PAULO	914	M	36900	47300
15	6433	EMILIA	SAO PAULO	914	F	79300	104500
16	6648	CLARA	RIO	918	F	24500	34100
17	6927	EDNA	RIO	914	F	27200	38500
18	7718	EDGAR	SAO PAULO	918	M	22400	31000
19	8113	JOSEFA	RIO	918	F	23800	37200
20	8123	SERGIO	RIO	918	M	95000	127200
21	8846	LUCIA	SAO PAULO	914	F	52100	83400
22	8989	ANTONIO	RIO	914	M	45500	65000

O SISTEMA E' ATIVADO ATRAVES DE UM CLIST, QUE DEFINE OS ARQUIVOS DE TRABAHO E CHAMA O MODULO S&S BASE PARA PROCESSAR O PROGRAMA EM MACRO LANGUAGE.

```

PROC 0 OPTIONS( ) ENTRY(SAS) OUTPLT(*) WORK(10) BLOCK(141)
CONTROL NOMSG
FREE F(PGM WORK TERML FT12F001 FT22F001 SASHELP SASAUTOS)
FREE ATTRLIST(S1)
DELETE S123.SAS
ATTR S1 RECFM(IV E A) LRECL(137) BLKSIZE(141)
ALLOC F(PGM) DAITABSAS.PGM)
ALLOCF(FT22F001) BLOCK(1024) SPACE(40 10) USING(S1)
ALLOCDAS(S123.SAS) SP(40 10) TR DIR(5) USING(S1)
ALLOCF(SASHELP) DAS(S123.SAS) OLD
ALLOCF(WORK) BLOCK(6394) SPACE(&WORK. 10)
ALLOCF(FT12F001) DA(*)
ALLOCF(ITERML) DA(*)
ALLOCF(SASAUTOS) DA('CV888.SAS.V516.MACAUTOS') SHR
SASCP TASKLIB('CV888.SAS.V516.LIBRARY' 'CV888.SAS.V516.LIBRARY2') +
ENTRY(SAS) OPTIONS('SYSIN=PGM NOCENTER NCNOTES CAPS')
END

```

O PONTO DE PARTIDA E' O MENU PRIMARIO COM AS OPCOES :

MENU PRIMARIO

1	=	DEFINICAO DO SAS DATA SET
2	=	MANUTENCAO
3	=	CONSULTA
X	=	FIM

=== DIGITE SUA OPCAO ===

1. OPCAD NO. 1 :

DEFINICAO DO SAS DATA SET

DSNAME DO ARQUIVO ==>

FOLHA.SAS

NOME DO SAS DATA SET ==>

CAD

NOME DO CAMPO CHAVE ==>

NFUN

2. CFCAC NO. 2 :

MANUTENCAO

1	=	INCLUIR REGISTROS
2	=	EXCLUIR REGISTROS
3	=	CORRIGIR REGISTROS
X	=	FIM

=== DIGITE SUA OPCAD ===

2.1.

INCLUIR REGISTRO

DIGITE O VALOR PARA O CAMPO NFUN ==>

11358

INCLUIR REGISTRO

CAMPO NOME ==>
HELIO

CAMPO CIDADE ==>
SAO PAULO

CAMPO DEPT ==>
914

CAMPO SEXO ==>
M

CAMPO SALLIQ ==>
110000

CAMPO SALBAS ==>
150000

REGISTRO INCLUIDO

NFUN=11358 NOME=HELIO CIDADE=SAO PAULO DEPT=914 SEXO=M
SALLIQ=110000 SALBAS=150000 _ERROR_=0 _N_=1

2.2.

EXCLUIR REGISTRO

DIGITE O VALOR PARA O CAMPO NFUN ==>

8989

VALOR ATUAL DO REGISTRO

NFUN=8989 NOME=ANTONIO CIDADE=RIO DEPT=914 SEXO=M SALLIQ=45500
SALBAS=65000 _ERROR_=0 _N_=11

DIGITE S PARA CONFIRMAR A EXCLUSAO OU N ==>

S

2.3.

CORRIGIR REGISTRO

DIGITE O VALOR PARA O CAMPO NFUN ==>

11400

VALOR ATUAL DO REGISTRO

NFUN=11400 NOME=MARIA CIDADE=SAO PAULO DEPT=918 SEXO=F
SALLIQ=20000 SALBAS=30000 _ERROR_=0 _N_=8

DIGITE A ALTERACAO NA FORMA VARIAVEL = VALOR
NO CASO DE MAIS DE UM CAMPO SEPARE POR
PONTO E VIRGULA

NOME = 'MARIANA'

VALOR ATUAL DO REGISTRO

NFUN=11400 NOME=MARIANA CIDADE=SAO PAULO DEPT=918 SEXO=F
SALLIQ=20000 SALBAS=30000 _ERROR_=0 _N_=8

2. OPCAO NO. 3 :

CONSULTA

DIGITAR O CRITERIO PARA A SELECAO
NA FORMA DE UMA EXPRESSAO LOGICA ==>

SEXO EQ F AND DEPT EQ 918

III. DESCRICAO DO PROGRAMA

O MENU PRIMARIO E' CBTIDO ATRAVES DA SEQUENCIA DE COMANDOS %PUT ABAIXO :

```
%DO I = 1 %TO 5;
```

```
  %PUT %STR( );
```

```
%END;
```

```
%PUT %STR(
```

```
) MENU PRIMARIO ;
```

```
%PUT %STR(
```

```
%PUT %STR(
```

```
%PUT %STR(
```

```
%PUT %STR(
```

```
%PUT %STR(
```

```
%PUT %STR(
```

```
%PUT %STR(
```

```
%PUT %STR( );
```

```
%PUT %STR(
```

1	=	DEFINICAO DO SAS DATA SET
2	=	MANUTENCAO
3	=	CONSULTA
X	=	FIM

```
) == DIGITE SUA OPCAO ==;
```

1. DEFINICAO DO SAS DATA SET

```
%INPUT DSN;
```

```
%TSO FREE F(DD1);
```

```
%TSO ALLOC CA(&CSN) F(DD1);
```

A MACRO VARIÁVEL &CSN É USADA PARA DEFINIR O SAS DATA SET, E OS COMANDOS %TSO PARA TORNA-LO DISPONÍVEL PARA O PROGRAMA.

A PROC CONTENTS SOBRE O DATA SET :

```
PROC CONTENTS DATA=CD1._ALL_ NOPRINT OUT=OUT1;RUN;
```

TEM COMO SAIDA O SAS DATA SET OUT1 :

OBSES	LIBRARY	MEMBER	MEMTYPE	NAME	TYPE	LENGTH	VAR	LABEL	FORMAT	FORMATT	FORMAT	INFORM	INFORM	INFORM	J	N	N
1	DD1	CAD		CIDADE	2	9	3		0	0		0	0	0	20	21	
2	DD1	CAD		DEPT	1	8	4		0	0		0	0	1	29	21	
3	DD1	CAD		NFUN	1	8	1		0	0		0	0	1	4	21	
4	DD1	CAD		NOME	2	8	2		0	C		0	0	0	12	21	
5	DD1	CAD		SALBAS	1	8	7		0	0		0	0	1	53	21	
6	DD1	CAD		SALLIQ	1	8	6		0	0		0	0	1	45	21	
7	DD1	CAD		SEXO	2	8	5		0	0		0	0	0	37	21	

ONDE SERA TESTADA A EXISTENCIA DO MEMBRO DEFINIDO PELO USUARIO :

```
%PUT NOME DO SAS DATA SET ==>;
%INPUT MEM;
%LET FLAG = N;
DATA _NULL_;
  SET OUT1;
  MEM1 = SYMGET('MEM');
  IF MEMNAME EQ UPCASE(MEM1) THEN DO;
    CALL SYMPUT('FLAG','S');
  STOP;
END;
%IF &FLAG NE S %THEN %DO;
%PUT NAO EXISTE O DATA SET = &MEM;
%GOTO LAB1;
%END;
```

A FUNCAO SYMGET ATRIEUI A VARIAVEL MEM1 O VALOR DA MACROVARIAVEL MEM PARA SER COMPARADA COM A VARIAVEL MEMNAME DE OUT1. SE O TESTE FOR VERDADEIRO A FUNCAO SYMPUT ATRIBUI A MACROVARIAVEL FLAG O CARACTER 'S'. APOS O STEP DATA A MACROVARIAVEL FLAG E TESTADA .

NA SEQUENCIA A PRCC CONTENTS SOBRE O MEMBRO

```
PRCC CONTENTS DATA=CC1.&MEM NOPRINT CUT=OUT1;RUN;
```

GERA O SAS DATA SET OUT1 :

	L	M	M	M		L	F	F	F	I	I	I				
	I	E	M	E		E	L	O	O	F	N	N				
	B	M	L	M		N	V	A	R	M	M	R	O	O	J	
	N	N	A	T		A	A	B	M	A	A	M	R	R	U	
	A	A	B	Y		A	Y	G	A	B	M	A	A	M	R	
	M	M	E	P		M	P	T	R	E	A	T	T	A	M	
	E	E	L	E		E	E	H	O	L	T	L	D	T	L	
	S															
1	DD1	CAD			CIDADE	2	9	3		0	0	0	0	0	20	21
2	DD1	CAD			DEPT	1	8	4		0	0	0	0	1	29	21
3	DD1	CAD			NFUN	1	8	1		0	0	0	0	1	4	21
4	DD1	CAD			NOME	2	8	2		0	0	0	0	0	12	21
5	DD1	CAD			SALEAS	1	8	7		0	0	0	0	1	53	21
6	DD1	CAD			SALLIQ	1	8	6		0	0	0	0	1	45	21
7	DD1	CAD			SEXC	2	8	5		0	0	0	0	0	37	21

A SEGUIR :

```
%LAB15:
  %PUT %STR( ); %PUT %STR( );
  %PUT  NOME DO CAMPO (HAVE ==>);
  %INPUT ;
  %LET CHV = %UPCASE(&SYSBUFFER);
  %LET FLAG = N;
  DATA _NULL_;
    SET CUT1;
    CALL SYMPUT('NVAR',_N_);
    AUX = 'VAR' || LEFT(VARO);
    CALL SYMPUT(AUX,NAME);
    AUX = 'T' || LEFT(VARO);
    CALL SYMPUT(AUX,TYPE);
    IF NAME EQ SYMGET('CHV') THEN CALL SYMPUT('FLAG','S');
  RUN;
  %IF &FLAG = N %THEN
  %DO;
    %PUT %STR( ); %PUT %STR( );
    %PUT  **** C CAMPO &CHV NAO EXISTE **** ;
    %GOTO LAB15;
  %END;
  %ELSE %GO TO LAB1;
  %END;
```

NO STEP DATA , A PARTIR DO ARQUIVO CUT1 SAO CRIADAS AS MACROVARIABLES :

NVAR = NUMERO DE VARIAVEIS NO DATA SET

VARI...VAR(NVAR) = VARIAVEIS DO SAS DATA SET

T1...T(NVAR) = TIPO DAS VARIAVEIS

C CAMPO CHAVE E' TESTADO COM A VARIABEL NAME .

2. MANUTENCAO

2.1. INCLUIR REGISTR

PARA CADA REGISTRO A SER INCLUIDO E' FEITO O TESTE PARA VERIFICAR SE JA EXISTE UM REGISTRO COM AQUELE VALOR DE CHAVE :

```

%PUT DIGITE O VALOR PARA O CAMPO &CHV ==>;
%INPUT VCHV;
%LET FLAG = N;
DATA _NULL_;
  SET DD1.&MEM;
  IF &CHV = SYMGET('VCHV') THEN DO;
    CALL SYMPUT('FLAG','S');
    PUT 'REGISTRO EXISTENTE';
    PUT _ALL_;
  END;
RUN;
%IF &FLAG EQ S %THEN %GOTO LAB3;

```

SE NAO EXISTIR, E' CRIADO UM DATA TEMPORARIO CONTENDO O REGISTRO. NO GRUPO %DO VARIANDO DE 1 ATE O NUMERO DE VARIAVEIS DO SAS DATA SET, TEMOS :

- A) PARA A VARIAVEL QUE FOI DEFINIDA COMO CAMPO CHAVE : E' ATRIBUINDO A ELA O VALOR DA MACROVARIAVEL &VCHV, DE ACORDO COM O SEU TIPO (NUMERICO OU CARACTER). DESVIA PARA O ENDERECC LF1.
- B) PARA AS DEMAIS VARIAVEIS: EXIBE NA TELA O NOME DO CAMPO PARA O USUARIO ATRIBUIR VALOR. LE ESSE VALOR COM O COMANDO INPUT ADEQUADO AO TIPO.

```

DATA TEMP;
  INFILE TERML ;
  %DO I = 1 %TO &NVAR;
    %IF &&VARI = &CHV %THEN %DO;
      %IF &&T&I = 1 %THEN %DO;
        LENGTH &CHV 8;
        &CHV = &VCHV;
        %END;
      %ELSE &CHV = SYMGET('VCHV');
      %GOTO LF1;
    %END;
    PUT " CAMPO &&VARI ==>";
    %IF &&T&I = 1 %THEN %DO;
      INPUT &&VARI ;
    %END;
    %ELSE %DO;
      INPUT &&VARI $ 1-72;
    %END;
  %LF1;
%END;

```

APÓS O GRUPO %DC, O REGISTRO É GRAVADO E EXIBIDO NA TELA.

```
CUTPUT;
PUT //;
PUT 'REGISTRO INCLUIDO';
PUT '-----';
PUT _ALL_;
IF _N_ = 1 THEN ^STOP;
RUN;
```

O REGISTRO É INCLUIDO NO SAS DATA SET POR UM COMANDO SET :

```
PRCC SORT DATA=DD1.&MEM; BY &CHV;
DATA DD1.&MEM ; SET DD1.&MEM TEMPA;
BY &CHV;
RUN;
```

2.2. EXCLUIR REGISTRO

O USUÁRIO FORNECE O VALOR DO CAMPO CHAVE DO REGISTRO A SER EXCLUÍDO. SE O REGISTRO EXISTIR, ELE É EXIBIDO, PARA QUE O USUÁRIO CONFIRME A EXCLUSÃO.

```
%PUT DIGITE O VALOR PARA O CAMPO &CHV ==>;
%INPUT VCHV;
%LET FLAG = N;
DATA _NULL_;
  SET DD1.&MEM;
  IF &CHV = SYMGET('VCHV') THEN DO;
    CALL SYMPUT('FLAG','S');
    PUT //;
    PUT _ALL_ //;
  END;
RUN;
%IF &FLAG NE S %THEN %DO;
  %PUT REGISTRO NÃO EXISTE ==> &CHV = &VCHV;
  %GOTO LAB3;
%END;
%PUT DIGITE S PARA CONFIRMAR A EXCLUSÃO OU N ==>;
%INPUT RESP;
```

SE FOR CONFIRMADA, O REGISTRO É EXCLUÍDO :

```
%IF %UPCASE(&RESP) = S %THEN %DO;
  DATA DD1.&MEM ;
  SET DD1.&MEM;
  IF &CHV = SYMGET('VCHV') THEN DELETE;
RUN;
%END;
```

2.3. CORRIGIR REGISTRO

ANALOGAMENTE AS OPERACOES DE INCLUSAO E EXCLUSAO, O REGISTRO A SER CORRIGIDO, E SELECIONADO PELO VALOR DE SEU CAMPO CHAVE. A ALTERACAO ENTRA NA FORMA DE UM COMANDO DE ATRIBUICAO VARIAVEL = VALOR. SE HOUVER MAIS DO QUE UMA ALTERACAO PARA O MESMO REGISTRO, ELAS DEVEM SER SEPARADAS POR PONTO E VIRGULA.

```

%INPUT ;
%LET STMT = %SYSBUFFER;
DATA DD1.%MEM ;
  SET DD1.%MEM;
  IF %CHV = SYMGET('VCHV') THEN DO;
    %STMT;
    PUT //;
    PUT 'VALOR ATUAL DO REGISTRO';
    PUT '-----';
    PUT _ALL_ ;
  END;

```

3. CONSULTA

A PARTIR DE UMA EXPRESSAO LOGICA, DEFINIDA PELO USUARIO CRIA-SE UM SUBSET DE REGISTROS :

```

%INPUT;
DATA DCONS;
  SET DD1.%MEM;
  IF %SYSBLFFR;
RUN;

```

A PROC PRINTTO E USADA PARA DIRECIONAR A SAIDA DA PROC PRINT, PARA UM ARQUIVO INTERMEDIARIC, NA UNIDADE 22.

```

PROC PRINTTO UNIT=22 NEW;
PROC PRINT DATA=DCONS UNIFORM SPLIT=' ';
PROC PRINTTO ;

```

O ARQUIVO DA UNIDADE 22 E ENTAO COPIADO PARA UM MEMBRO, DE NOME SAIDA, DE UM PARTICIPADO, QUE ESTA ALLOCADO COMO FILE SASHELP :

```

DATA _NULL_;
  INFILE FT22F001;
  INPUT;
  FILE SASHELP(SAIDA) NOPRINT;
  PUT _INFILE_;
RUN;

```

O COMANDO HELP E USADO PARA QUE O CONTEUDO DE SAIDA SEJA EXIBIDO EM TELA DO TIPO "FULL SCREEN" :

```

HELP SAIDA;
%END;

```