

# Programação para Mainframe



**Curso de Análise de Desenvolvimento de Sistemas**

**Aula 07**

**Prof. Claudio Benossi**



CREATE

READ

UPDATE

DELETE

C

R

U

D

# Introdução ao COBOL

---

Nosso objetivo hoje é criar CRUD, um sistema que permita controlar o estoque de produtos, neste sistemas vamos criar um arquivo para inserir, consultar, alterar e deletar os de diferentes produtos.





# Introdução ao COBOL

## 1º Passo:

Fazer a indicação do arquivo que será usados como fonte dos dados, no caso o arquivo **PRODUTOS** 'produtos.txt'.

4

## PRODUTOS

COD-PRODUTO	NOME-PRODUTO	ESTOQUE-PRODUTO
PIC 9(003)	PIC X(020)	PIC 9(009)





# Introdução ao COBOL

Para essa atividade, vamos criar as operações em arquivos separados, primeiro o programa responsável pela inclusão dos dados. (No futuro vamos fazer tudo em um único arquivo e depois também vamos fazer a união de vários arquivos como módulos de um sistema).



# Introdução ao COBOL

## 2º Passo:

Na segunda divisão do Cobol na ENVIRONMENT DIVISION, dentro da INPUT-OUTPUT SECTION, no FILE-CONTROL, vamos fazer a indicação do nosso arquivo, e agora vamos começar a trabalhar com arquivos organizados de forma indexada e com o uso de chave primária.

```
FILE-CONTROL.  
  SELECT PRODUTOS ASSIGN TO  
    'C:\programas\ARQUIVOS\produtos.txt'  
  ORGANIZATION IS INDEXED  
  ACCESS IS RANDOM  
  RECORD KEY IS COD-PRODUTO  
  FILE STATUS IS WS-FS.
```

```
DATA DIVISION.  
FILE SECTION.  
FD PRODUTOS.  
  01 REG-PRODUTO.  
    03 COD-PRODUTO          PIC 9(03).  
    03 NOME-PRODUTO         PIC X(20).  
    03 ESTOQUE-PRODUTO      PIC 9(09).
```

# Introdução ao COBOL

## 3º Passo:

Ainda na DATA DIVISION, dentro da WORKING-STORAGE SECTION criar as variáveis que serão utilizadas durante a execução do programa, como nesse primeiro programa vamos fazer somente a inserção dos dados dos produtos, vamos usar somente uma variável de apoio que será usada para armazenar os dados do File Status do nosso arquivo de PRODUTOS.

```
WORKING-STORAGE SECTION.  
77 WS-FS                      PIC 99.
```

# Introdução ao COBOL

## 4º Passo:

Agora na PROCEDURE DIVISION, vamos por em prática a nossa lógica.

- Vamos iniciar a variável de File Status e abrir o arquivos de acordo com o que se pretende fazer com o mesmo:

```
PROCEDURE DIVISION.  
MAIN-PROCEDURE.  
    DISPLAY "----- CADASTRO DE PRODUTOS -----"  
    SET WS-FS      TO 0.  
    OPEN I-O PRODUTOS  
    IF WS-FS EQUAL 35 THEN  
        OPEN OUTPUT PRODUTOS  
    END-IF
```





# Introdução ao COBOL

5º Passo:

Agora que vamos verificar se o File Status está ok e vamos solicitar ao usuário que insira os dados via teclado:

```
IF WS-FS EQUAL ZEROS  
DISPLAY 'INFORME O CODIGO DO PRODUTO:'  
ACCEPT COD-PRODUTO  
DISPLAY 'INFORME O NOME DO PRODUTO'  
ACCEPT NOME-PRODUTO  
DISPLAY 'INFORME A QUANTIDADE EM ESTOQUE DO PRODUTO'  
ACCEPT ESTOQUE-PRODUTO
```





# Introdução ao COBOL

## 6º Passo:

Após a solicitação dos dados via comando DISPLAY e recuperar os dados informados via teclado pelo comando ACCEPT e inseridos diretamente nos campos do nosso registro, precisamos solicitar a gravação desse registro:

10

```
*****  
COMANDO PARA INSERIR DADOS NO ARQUIVO *****  
WRITE REG-PRODUTO
```



# Introdução ao COBOL

7º Passo:

Agora que já salvamos as informações no arquivo, vamos fazer uma verificação se tudo ocorreu de forma correta.

```
***** VERIFICANDO SE O REGISTRO FOI REALIZADO *****  
IF WS-FS NOT EQUAL ZEROS  
    DISPLAY 'ERRO - NÃO FOI POSSIVEL GRAVAR O REGISTRO'  
    DISPLAY 'FILE STATUS: ' WS-FS  
ELSE  
    DISPLAY 'REGISTRO GRAVADO COM SUCESSO!'  
END-IF
```



# Introdução ao COBOL

8º Passo:

No 5º Passo, fizemos uma verificação antes de solicitar os dados via teclado e agora vamos fazer o SENÃO (caso o File Status apresente algum erro:

```
ELSE  
    DISPLAY 'ERRO AO CRIAR O ARQUIVO'  
    DISPLAY 'FILE STATUS: ' WS-FS  
END-IF
```





# Introdução ao COBOL

9º Passo:

Para finalizar nosso processo de inclusão, vamos fechar o nosso arquivo:

CLOSE PRODUTOS



```
1 *****
2 * Author: CLAUDIO BENOSSE
3 * Date: 06/09/2021
4 * Purpose: TRABALHANDO COM ARQUIVOS DE DADOS INDEXADOS - INCLUIR
5 * Tectonics: cobc
6 *****
7 IDENTIFICATION DIVISION.
8 PROGRAM-ID. PROGRAMA_EX13.
9 ENVIRONMENT DIVISION.
10 INPUT-OUTPUT SECTION.
11 FILE-CONTROL.
12 SELECT PRODUTOS ASSIGN TO
13     'C:\programas\ARQUIVOS\produtos.txt'
14 ORGANIZATION IS INDEXED
15 ACCESS IS RANDOM
16 RECORD KEY IS COD-PRODUTO
17 FILE STATUS IS WS-FS.
18 DATA DIVISION.
19 FILE SECTION.
20 FD PRODUTOS.
21 01 REG-PRODUTO.
22     03 COD-PRODUTO          PIC 9(03).
23     03 NOME-PRODUTO         PIC X(20).
24     03 ESTOQUE-PRODUTO      PIC 9(09).
25 WORKING-STORAGE SECTION.
26 77 WS-FS                   PIC 99.
27 PROCEDURE DIVISION.
```

```
27✓ PROCEDURE DIVISION.
28✓ MAIN-PROCEDURE.
29   DISPLAY "----- CADASTRO DE PRODUTOS -----"
30   SET WS-FS      TO 0.
31   OPEN I-O PRODUTOS
32✓   IF WS-FS EQUAL 35 THEN
33       OPEN OUTPUT PRODUTOS
34   END-IF
35   IF WS-FS EQUAL ZEROS
36       DISPLAY 'INFORME O CODIGO DO PRODUTO:'
37       ACCEPT COD-PRODUTO
38       DISPLAY 'INFORME O NOME DO PRODUTO'
39       ACCEPT NOME-PRODUTO
40       DISPLAY 'INFORME A QUANTIDADE EM ESTOQUE DO PRODUTO'
41       ACCEPT ESTOQUE-PRODUTO
42   ***** COMANDO PARA INSERIR DADOS NO ARQUIVO *****
43   WRITE REG-PRODUTO
44   ***** VERIFICANDO SE O REGISTRO FOI REALIZADO *****
45✓   IF WS-FS NOT EQUAL ZEROS
46       DISPLAY 'ERRO - NÃO FOI POSSIVEL GRAVAR O REGISTRO'
47       DISPLAY 'FILE STATUS: ' WS-FS
48✓   ELSE
49       DISPLAY 'REGISTRO GRAVADO COM SUCESSO!'
50   END-IF
51✓   ELSE
52       DISPLAY 'ERRO AO CRIAR O ARQUIVO'
53       DISPLAY 'FILE STATUS: ' WS-FS
54   END-IF
55   CLOSE PRODUTOS
56   STOP RUN.

END PROGRAM PROGRAMA_EX13
```

# Introdução ao COBOL

OK funcionou!





# Introdução ao COBOL

**DESAFIO**



Nosso programa está funcionando, porém pode ser melhorado, quando quero inserir mais de um produto, preciso executar novamente, gostaria de poder inserir mais de um produto por vez e também gostaria de ter uma validação dos dados antes de inserir no arquivo para evitar problemas futuros.

COD-PRODUTO	NOME-PRODUTO	ESTOQUE-PRODUTO
PIC 9(003)	PIC X(020)	PIC 9(009)



# Introdução ao COBOL

Agora vamos criar nosso segundo arquivo do nosso CRUD, um arquivo para fazer a listagem de todos os produtos cadastrados.

Lembrando que esse programa tem como objetivo somente consultar os dados, podemos abrir nosso arquivo como INPUT.



# Introdução ao COBOL

```
ENVIRONMENT DIVISION.  
INPUT-OUTPUT SECTION.  
FILE-CONTROL.  
    SELECT PRODUTOS ASSIGN TO  
        'C:\programas\ARQUIVOS\produtos.txt'  
    ORGANIZATION IS INDEXED  
    ACCESS IS DYNAMIC  
    RECORD KEY IS COD-PRODUTO  
    FILE STATUS IS WS-FS.
```

```
DATA DIVISION.  
FILE SECTION.  
FD PRODUTOS.  
01 REG-PRODUTO.  
    03 COD-PRODUTO          PIC 9(03).  
    03 NOME-PRODUTO         PIC X(20).  
    03 ESTOQUE-PRODUTO      PIC 9(09).
```



# Introdução ao COBOL

Outro aspecto importante será as variáveis utilizadas nesse processo, precisamos agora criar um espelho do arquivo produto, no caso um WS-PRODUTO, também algumas variáveis de apoio, como uma variável para controlar a quantidade de registros inseridos WS-CONTADOR, e uma variável para controlar o processo de leitura do arquivo para indicar se é o final do arquivo WS-EOF.



# Introdução ao COBOL

```
WORKING-STORAGE SECTION.
```

```
77 WS-FS          PIC 99.
```

```
77 WS-EOF         PIC 99.
```

```
77 WS-CONTADOR    PIC 99.
```

```
01 WS-PRODUTO.
```

```
    03 WS-COD-PRODUTO          PIC 9(03).
```

```
    03 WS-NOME-PRODUTO         PIC X(20).
```

```
    03 WS-ESTOQUE-PRODUTO      PIC 9(09).
```

# Introdução ao COBOL

Vamos fazer a leitura do arquivo e exibir os dados:

```
PROCEDURE DIVISION.  
MAIN-PROCEDURE.  
    OPEN INPUT PRODUTOS  
    DISPLAY '----- LISTAGEM DE PRODUTOS -----'  
    SET WS-EOF      TO 0  
    SET WS-FS       TO 0  
  
    PERFORM UNTIL WS-EOF EQUAL 1  
        READ PRODUTOS INTO WS-PRODUTO  
        AT END  
            MOVE 1 TO WS-EOF  
        NOT AT END  
            DISPLAY ' COD. PRODUTO: ' WS-COD-PRODUTO  
                  ' NOME PRODUTO: ' WS-NOME-PRODUTO  
                  ' QTDE EM ESTOQUE: ' WS-ESTOQUE-PRODUTO  
    END-READ  
END-PERFORM
```



# Introdução ao COBOL

Não esqueça de fechar o seu arquivo:

```
CLOSE PRODUTOS
```

```
STOP RUN.
```



```

1      *****
2      * Author: CLAUDIO BENOSSE
3      * Date: 06/09/2021
4      * Purpose: TRABALHANDO COM ARQUIVOS DE DADOS INDEXADOS - LISTAR
5      * Tectonics: cobc
6      *****
7      IDENTIFICATION DIVISION.
8      PROGRAM-ID. PROGRAMA_EX17.
9      ENVIRONMENT DIVISION.
10     INPUT-OUTPUT SECTION.
11     FILE-CONTROL.
12         SELECT PRODUTOS ASSIGN TO
13             'C:\programas\ARQUIVOS\produtos.txt'
14         ORGANIZATION IS INDEXED
15         ACCESS IS DYNAMIC
16         RECORD KEY IS COD-PRODUTO
17         FILE STATUS IS WS-FS.
18     DATA DIVISION.
19     FILE SECTION.
20     FD PRODUTOS.
21     01 REG-PRODUTO.
22         03 COD-PRODUTO          PIC 9(03).
23         03 NOME-PRODUTO         PIC X(20).
24         03 ESTOQUE-PRODUTO      PIC 9(09).

```



```
25~ WORKING-STORAGE SECTION.
26 77 WS-FS PIC 99.
27 77 WS-EOF PIC 99.
28 77 WS-CONTADOR PIC 99.
29
30~ 01 WS-PRODUTO.
31 03 WS-COD-PRODUTO PIC 9(03).
32 03 WS-NOME-PRODUTO PIC X(20).
33 03 WS-ESTOQUE-PRODUTO PIC 9(09).
34
35~ PROCEDURE DIVISION.
36~ MAIN-PROCEDURE.
37 OPEN INPUT PRODUTOS
38 DISPLAY '----- LISTAGEM DE PRODUTOS -----'
39 SET WS-EOF TO 0
40 SET WS-FS TO 0
41 SET WS-CONTADOR TO 0
42~ PERFORM UNTIL WS-EOF EQUAL 1
43~ READ PRODUTOS INTO WS-PRODUTO
44~ AT END
45 MOVE 1 TO WS-EOF
46~ NOT AT END
47~ DISPLAY ' COD. PRODUTO: ' WS-COD-PRODUTO
48 ' NOME PRODUTO: ' WS-NOME-PRODUTO
49 ' QTDE EM ESTOQUE: ' WS-ESTOQUE-PRODUTO
50 ADD 1 TO WS-CONTADOR
51 END-READ
52 END-PERFORM
53 DISPLAY 'TOTAL DE REGISTRO LOCALIZADOS: ' WS-CONTADOR
54 CLOSE PRODUTOS
55 STOP RUN.
56 END PROGRAM PROGRAMA_EX17.
```

# Introdução ao COBOL



OK, porém vamos melhorar esse programa.

Exemplo: Caso não tenha nenhum registro - informar  
“Nenhum Registro localizado!”.





# Introdução ao COBOL

OK nosso segundo programa do nosso sistema de estoque, faz a listagem de todos os produtos cadastrados, porém vamos desenvolver um terceiro programa para fazer a consulta de um produto específico, através do seu código de produto:



# Introdução ao COBOL

1º Passo:

Identificação do arquivo:

```
IDENTIFICATION DIVISION.  
PROGRAM-ID. PROGRAMA_EX14.  
ENVIRONMENT DIVISION.  
INPUT-OUTPUT SECTION.  
FILE-CONTROL.  
    SELECT PRODUTOS ASSIGN TO  
        'C:\programas\ARQUIVOS\produtos.txt'  
    ORGANIZATION IS INDEXED  
    ACCESS IS RANDOM  
    RECORD KEY IS COD-PRODUTO  
    FILE STATUS IS WS-FS.  
DATA DIVISION.  
FILE SECTION.  
FD PRODUTOS.  
01 REG-PRODUTO.  
    03 COD-PRODUTO          PIC 9(03).  
    03 NOME-PRODUTO         PIC X(20).  
    03 ESTOQUE-PRODUTO      PIC 9(09).
```



# Introdução ao COBOL

2º Passo:

Criar as variáveis:

```
WORKING-STORAGE SECTION.  
77 WS-FS                PIC 99.  
  
01 WS-PRODUTO.  
   03 WS-COD-PRODUTO    PIC 9(03).  
   03 WS-NOME-PRODUTO   PIC X(20).  
   03 WS-ESTOQUE-PRODUTO PIC 9(09).
```





# Introdução ao COBOL

3º Passo:

Vamos solicitar ao usuário que informe qual o código do produto a ser pesquisado:

```
PROCEDURE DIVISION.  
  MAIN-PROCEDURE.  
    OPEN I-O PRODUTOS.  
    DISPLAY '----- CONSULTA DE PRODUTOS -----'  
    DISPLAY 'INFORME O CODIGO DO PRODUTO'  
    ACCEPT COD-PRODUTO
```





# Introdução ao COBOL

## 4º Passo:

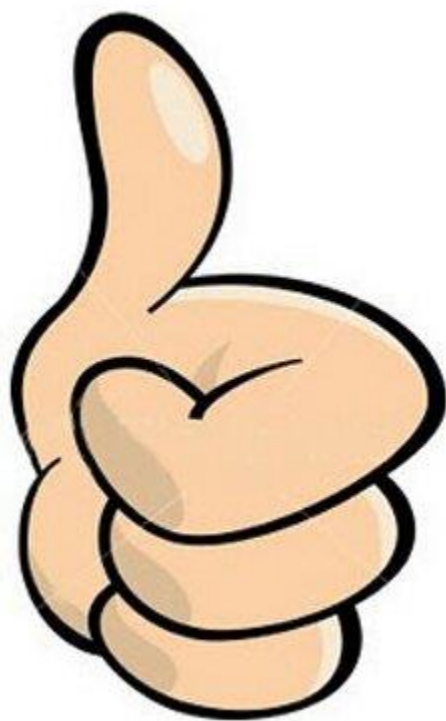
Vamos fazer a leitura do arquivo indicando como parâmetro de busca a chave primária que é o código do produto e exibir o resultado:

```
READ PRODUTOS RECORD INTO WS-PRODUTO
  KEY IS COD-PRODUTO
  INVALID KEY
    DISPLAY 'CODIGO DO PRODUTO INVALIDO'
  NOT INVALID KEY
    DISPLAY 'CODIGO DO PRODUTO: ' WS-COD-PRODUTO
    DISPLAY 'NOME DO PRODUTO: ' WS-NOME-PRODUTO
    DISPLAY 'QTDE EM ESTOQUE: ' WS-ESTOQUE-PRODUTO
END-READ.
CLOSE PRODUTOS.
```



# Introdução ao COBOL

OK funcionou!



**MAIS  
OU  
MENOS**





# Introdução ao COBOL

SUGESTÃO DE

MELHORIAS



OK, porém vamos melhorar esse programa.

Exemplo:

- Caso não tenha nenhum registro - informar “Nenhum Registro localizado!”.
- Verificar se o código informado pelo usuário é válido!
- ???

# Introdução ao COBOL



**DESAFIO**



Nosso pequeno sistema de Estoque está melhorando, agora gostaria de ter um programa que possa fazer a listagem de todos os produtos, mas também me ofereça a oportunidade de localizar um registro específico. (Unir os dois últimos programas desenvolvidos).

"Nunca entro em pânico  
quando me perco.

Eu só troco o lugar para  
onde quero ir."



**Rita Rudner**

# Obrigado!

**Se precisar ...**

Prof. Claudio Benossi

[claudio.benossi@fatec.sp.gov.br](mailto:claudio.benossi@fatec.sp.gov.br)

