

# Formação: COBOL

Professor: Vagner Bellacosa

Disciplina: Linguagem de Programação COBOL

# COBOL

## Modulo 06.01 – Uso avançado das Variáveis

- 1) Formato das variáveis
- 2) Variáveis
- 3) Formato Var. Numéricas
- 4) Formato Var. Alfanuméricas
- 5) Formato Var. Alfabéticas
- 6) Mascara edição alfa
- 7) Mascara edição numérica
- 8) Valor inicial.
- 9) Comando DISPLAY
- 10) Comando GOBACK
- 11) Comando STOP RUN

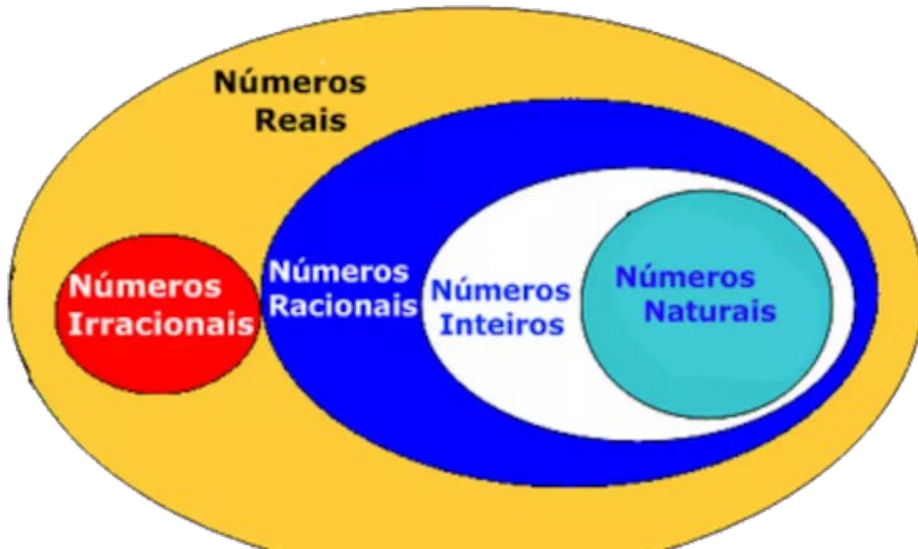


A Lógica é o princípio mais importante no desenvolvimento de programas de computador.

Todo aplicativo ou sistema de computador deve ser projetado e desenvolvido de forma racional para que possamos chegar a sequências lógicas de instruções que sejam coerentes e, principalmente, eficazes.

# COBOL

## Formato das variáveis



No modulo anterior vimos a palavra reservada PIC, diminutivo de PICTURE.

Com ela podemos definir o formato da variáveis, bem como seu tamanho e algumas regras de edição/exibição dos dados.

Lembrando que o COBOL permite trabalhar com números REAIS e por isso necessita de notação especial.

# COBOL

## Formato das variáveis numéricas

Para indicar numero fracionado, usamos a mascara **V**, a partir deste ponto estaremos na parte decimal do numero.

A mascara **S**, indica que é um numero com sinal, podendo ser positivo ou negativo.

Estes campos mascarados tem um limite de 18 dígitos

Podemos executar cálculos, porém a performance cai e não é uma pratica recomendada.

```
000001 *****
000002 ***  VARIAVEIS E SEUS FORMATOS PIC E MASK  ***
000003 *****
000004
000005 *  VARIAVEIS NUMERICAS
000006 10  ACM-TOTAL-VENDA          PIC 9(10).
000007 10  ACM-PRECO-VENDA        PIC 9(10)V99.
000008 10  ACM-RESULTADO          PIC S9(02).
000009
```

# COBOL

## Formato das variáveis alfanuméricas

Para indicar um texto alfanumérico, isto é, contendo letra, número e caracteres especiais.

Usamos a máscara **X**.

Estes campos mascarados têm um limite de 140 dígitos

Não podem ser feitos cálculos com esta variável.

```
000010 * VARIÁVEIS ALFANUMÉRICAS
000011 10 ACM-TOMBO-LIVRO          PIC X(10).
000012 *
```

# COBOL

## Formato das variáveis alfabéticas

Para indicar um texto alfabético

Usamos a mascara **A**.

Estes campos mascarados tem um limite de 140 dígitos

Não podem ser feitos cálculos com esta variável.

```
000012
000013 * VARIAVEIS ALFABETICAS
000014 10 ACM-NOME-LIVRO          PIC A(30).
000015
```

# COBOL

## Mascara de edição alfanumérica

Usamos a mascara **B** para incluir espaços em branco na string, usada para displays e impressão de relatórios.

Usamos o **0** para inserir zeros na mascara..

```

000016 * MASCARA EDICAO ALFANUMERICA
000017 10 REL-COD-TMB-BOOK      PIC 0X(02).
000018 10 REL-COD-MTZ-BOOK     PIC 0X(03)BX(02).
000019 10 REL-COD-Z-BOOK       PIC BBBX(04).
000020
                                EXEMPLO
                                *** 0AN
                                *** 0ANC 91
                                *** ABCD
    
```

# COBOL

## Mascara de edição numérica

No display de números e impressão de relatório existe uma gama de combinação de sinais para cobrir diversas formatações numéricas.

9 – Mostra o dígito na posição definida independente de ser “0”;  
Z – Troca o dígito “0” pelo caractere BRANCO;  
B – Insere o caractere BRANCO na posição definida na picture;  
0 – Coloca o dígito “0” na posição definida na picture;

, – Insere uma “,” na posição definida na picture;  
. – Insere um “.” na posição definida na picture;  
+ – Insere o sinal “+” ou “-” à direita ou à esquerda do conteúdo editado;  
- – Insere o sinal “-” à direita ou à esquerda do conteúdo editado;  
\* – Troca o “0” não significativo por “\*” (asterisco);  
\$ – Este caractere pode aparecer na extremidade esquerda da picture.



# COBOL

## Mascara de edição numérica

```

EDIT          KC02746.WORKBOOK.COBO(LVARMASK) - 01.10          Member VARMASK saved
Command ==>  _          Scroll ==>  CSR
000020
000021 * MASCARA EDICAO NUMERICA                                EXEMPLO
000022 10 REL-COD-MASK          PIC ZZZ          123
000023 10 REL-COD-PT-BR        PIC Z.ZZ9        1.900
000024 10 REL-COD-DEC-PT-BR    PIC Z.ZZ9,99     543,55
000025 10 REL-COD-BLOQ         PIC *99,99       *77,77
000026 10 REL-COD-SPACE        PIC 9B9          6 6
000027 10 REL-COD-FAKE         PIC 99,00        88,00
000028 10 REL-COD-SIGNAL1      PIC +ZZ9         +876
000029 10 REL-COD-SIGNAL2      PIC 99+          56-
000030 10 REL-COD-SIGNAL3      PIC 99+          56+
000031 10 REL-COD-SIGNAL4      PIC +99          +56
000032 10 REL-COD-SIGNAL5      PIC -99          -56
000033 10 REL-COD-SIGNAL6      PIC 99-          56-
000034 10 REL-COD-SIGNAL7      PIC -99          56
000035 10 REL-COD-MONEY        PIC $9(4)        $8765
000036

```

# COBOL

## Inicializando variáveis com VALUE

As vezes em nossa codificação precisamos inicializar uma variáveis com algum valor.

Para isso usamos a palavra reservada **VALUE**, que através dela podemos passar valor iniciais.

Se for alfanumérico ou alfabético deve estar contido entre aspas ou apostrofes.

Usamos as palavras reservadas **SPACE(S)**, **ZERO**, **ZEROES**, **ALL**

```
000036
000037 * INICIALIZANDO VARIÁVEIS
000038 10 ACM-TOTAL-NUM      PIC 9(04)      VALUE 1234.
000039 10 ACM-ESTADO        PIC X(02)      VALUE 'SP'.
000040 05 FILLER              PIC X(4)       VALUE SPACES.
000041 05 FILLER              PIC X(20)      VALUE ALL " * ".
000042 05 FILLER              PIC X(20)      VALUE.
000043 05 FILLER              PIC 9(3)       VALUE ZEROS.
```

# COBOL

## DISPLAY

Comando utilizado para comunicação com os usuários, serve para exibir strings e números no decorrer do processamento.

Neste primeiro momento vamos trabalhar com a sintaxe básica, posteriormente veremos modos avançados.

```
000001 *****
000002 ***  COMANDOS COBOL
000003 *****
000004
000005 *  DISPLAY
000006      DISPLAY 'TOTAL DE VENDAS :'  ACM-TOTAL-VENDA.
000007
000008      DISPLAY  ACM-RESULTADO.
000009
```

# COBOL

## Fim do Processamento

Existem duas maneiras de um programa COBOL, terminar:

- 1) Termino Normal, ocorrendo quando chegar no ultimo comando.
- 2) **ABEND**, ou *Abnormal END*, final anormal



# COBOL

## GOBACK

O comando **GOBACK** termina parcialmente o programa, pois em verdade, ele devolve o ponteiro da execução ao programa chamador.

Usado em rotinas online CICS e subrotinas chamadas por pelo comando **CALL**.

```
000012  
000013 * COMANDO GOBACK  
000014 GOBACK.  
000015
```

# COBOL

## EXIT PROGRAM

O comando **EXIT PROGRAM**, funciona de modo similar ao GOBACK.

Devendo ser usado em programas online CICS e SubRotinas chamadas via comando CALL.

Tem um comportamento bizarro se usado no modo BATCH.

```
000015  
000016 * COMANDO EXIT PROGRAM  
000017      EXIT PROGRAM.  
000018
```

# COBOL

## STOP RUN

O comando **STOP RUN**, termina todo e qualquer programa a ele associado.

Perigoso, pois pode derrubar o CICS.

Este comando deve ser utilizado apenas em programas Batch.

```
000009  
000010 * COMANDO STOP RUN  
000011      STOP RUN.  
000012
```

# COBOL

## BÔNUS: RETURN-CODE

O registro especial RETURN-CODE tem a definição implícita.

```
01 RETURN-CODE GLOBAL PICTURE S9(4) USAGE BINARY VALUE ZERO.
```

Usado para devolver ao SDSF, um código de retorno, normalmente indicando um ABEND programado, dentro as validações do programa.

Sendo que o operador deverá consultar o manual do programa, para tomar as providências.

- COMPUTE RETURN-CODE = 8.
- MOVE 8 to RETURN-CODE.



# COBOL

Duvidas?

