

Capítulo 3 : Comandos e instruções



Aviso

Todos os ensinamentos foram adquiridos por livros , PDFs e cursos ; créditos apresentados no arquivo README.



Aviso

Exemplos de códigos se encontraram nessa pasta com o nome de COMANDOS.

A melhor forma de começar a estudar comandos e suas funcionalidades dentro de um programa COBOL, primeiro devemos compreender qual a definição de comandos e instruções.

Comandos nada mais são do que ordens de execução de processamento , já instruções são ações de manipulação de processamentos. Dando um contexto como exemplo, quando estamos realizando uma atribuição , uma movimentação, um acesso ao banco de dados , realizamos manipulação de arquivos, em outras palavras , toda e qualquer ordem dada ao programa se trata de um comando , já ações imperativas se trata de instruções.

Comandos de encerramento

Alguns comandos de encerramento devem ser ajustados de acordo com a necessidade pois um encerramento de subprograma , pode comprometer o funcionamento de um programa principal, por exemplo.

Dentro do desenvolvimento COBOL , possuímos duas categorias , os programas e os relatórios. Há uma diferença bem sutil entre eles.

Programa		Relatório
Principal	Subprograma (sub-rotina) (módulo)	
Comando STOP RUN. Comando GOBACK.	Comando GOBACK. Comando EXIT PROGRAM. Comando CANCEL.	Comando TERMINATE.
Comando GOBACK retorna de onde veio. Comando STOP RUN retorna para o sistema operacional da máquina.	Comando GOBACK retorna de onde veio. Comando EXIT PROGRAM utilizado apenas para subprogramas. Comando CANCEL aborta /anula qualquer processamento em execução.	Comando TERMINATE utilizado apenas para relatórios.

Logo após essa pequena introdução deve-se dizer que programas são um conjunto de funções , atribuições , etc. que são chamados , realizam o processamento e não necessariamente devem retornar algo á alguém.

Já relatórios são também funções , atribuições etc. que são chamadas , ou não , podendo ser executadas independentemente do programa principal , onde esse realiza o processamento e retorna uma saída tanto externa ou não. Em outras palavras ele pode enviar o resultado tanto para outros programas ou subprogramas , quanto também pode enviar para uma impressora por exemplo.

Comandos

Comando DISPLAY

O comando display é responsável por mostrar / visualizar / exibir dados para fora do programa. Esse comando possui o seguinte formato

- DISPLAY <literal>
- DISPLAY <comandos / <dados >
- DISPLAY <literal> <dados>

Comando ACCEPT

O comando responsável por pegar informações do sistema , em uma forma mais tranquila de entender para programadores Java, esse comando é o scan da linguagem , podendo ser informações essas vindas de um teclado ou diretamente do sistema seguindo a estrutura abaixo.

- ACCEPT <dados> FROM (DATE || TIME || DAY || SCAPE KEY)
- ACCEPT (L , C) <dados>
 - L = Linha
 - C = Coluna

Comando MOVE

Comando responsável por realizar atribuição de valores , conteúdos , etc. a variáveis.

Comando SET

Bastante semelhante com o comando MOVE , o comando SET é mais assertivo , sendo mais recomendado em variáveis booleanas.

A diferença é que quando declaramos uma variável no espaço de memória é criado esse ponto de referencia , quando utilizamos o MOVE , esse valor será direcionado para a variável e depois alterado no espaço de memória , então todas as demais variáveis que olham para esse ponto de referencia iram receber essa alteração , o comando SET impede essa alteração de variáveis indevidas.

Comando PERFORM

O comando PERFORM é um comando para navegar entre as PROCEDURES e parágrafos, comando esse realiza chamadas de parágrafos onde esses contem comandos e demais linhas de código . Segue um exemplo de sua estrutura do comando.

- PERFORM P1 THRU P1-FIM.

Também pode ser utilizado da seguinte forma , a divisão dos parágrafos em SECTIONs podendo ser chamados com o seguinte comando.

- PERFORM S1.
 - S1 SECTION.