

Estrutura de linguagem 2016.2

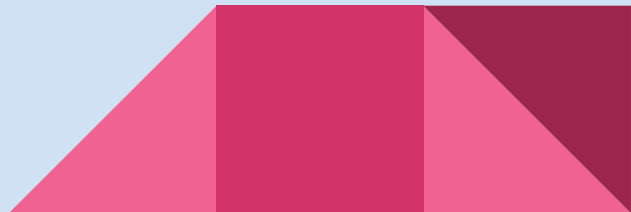
# Cobol

Por Maurício Vicente



# Origem

- COBOL foi desenvolvido para fornecer uma linguagem padrão em diferentes tipos de hardware, especificamente para a programação de negócios.
- Não haviam outras linguagens naquela época que funcionava desta forma para fins comerciais.
- Foi inspirada em grande parte pela linguagem **FLOW-MATIC** pela linguagem **COMTRAN**.
- O **COBOL** foi desenvolvido num período de seis meses e, mais de 40 anos depois, ainda é muito utilizada.



# Classificação

- Permite tipagem Forte e estática
  - O programador utiliza recurso de tipagem estática. Ao contrário do que acontece no **PHP**, pode-se declarar uma variável inteira da seguinte forma.

```
01 WS-CLASS    PIC 9(2)  VALUE  '10'.
```

- Expressividade
  - Em relação ao **PHP**, a linguagem cobol oferece uma série de dificuldade para programadores. Na leitura e na escrita não é expressivo pois a forma de escrever é rígida e não é intuitiva. A seguir o código comparando as duas linguagens.

# Comparação Cobol e Php

```
IDENTIFICATION DIVISION.  
PROGRAM-ID. HELLO.  
  
DATA DIVISION.  
    WORKING-STORAGE SECTION.  
    01 WS-Contador PIC 9(2) VALUE '1'.  
  
PROCEDURE DIVISION.  
    A-PARA.  
    PERFORM B-PARA VARYING WS-Contador FROM 1 BY 1 UNTIL WS-Contador=10  
    STOP RUN.  
  
    B-PARA.  
    DISPLAY 'Count: ' WS-Contador.
```

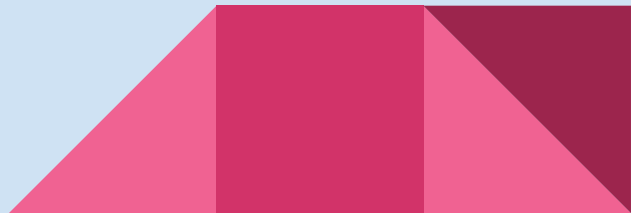
```
<?php  
$count = 1;  
  
while($count < 10)  
{  
    echo 'Count :' . $count;  
    $count+=1;  
}  
?>
```

```
Count :1  
Count :2  
Count :3  
Count :4  
Count :5  
Count :6  
Count :7  
Count :8  
Count :9
```

- Para o indexador de um tabela em cobol , o indexador não pode ser inicializada por 0.

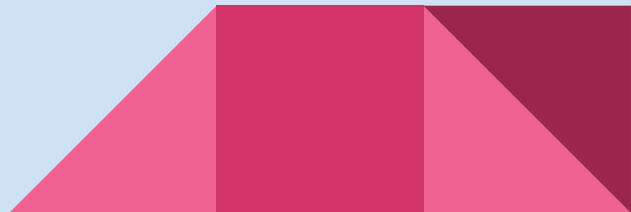
# Conclusão

- Tradicionalmente a COBOL é uma linguagem simples, com alcance limitado da função (sem ponteiros, sem tipos definidos pelo usuário e sem funções definidas pelo usuário) estimulando um estilo de codificação simples. Isso fez com que seja bem adequada ao seu domínio principal de computação de negócios, onde a complexidade do programa encontra-se em regras de negócio que precisam ser codificados em vez de sofisticados algoritmos e estruturas de dados.
- Porém a leitura da linguagem é complexa, não é possível saber o que o programa está fazendo sem ter uma grande prática com a linguagem e conhecimento da documentação.
- Somente um bom conhecimento da documentação não basta, ainda sim é difícil escrever o código pois são muito detalhes a lembrar.



# Bibliografia

- Cobol Wikipédia - <https://pt.wikipedia.org/wiki/COBOL>
- Cobol Ainda esta viva DevMedia - <http://www.devmedia.com.br/cobol-uma-linguagem-que-ainda-esta-viva/24585>
- Cobol Point - [https://www.tutorialspoint.com/pg/cobol/cobol\\_data\\_types.htm](https://www.tutorialspoint.com/pg/cobol/cobol_data_types.htm)
- Linguagem Cobol 1959 - <https://sites.google.com/site/linguagemcobol1959/home>



Fim

