

FATEC Santana de Parnaíba

Curso: Análise e Desenvolvimento de Sistemas

Disciplina: Linguagem de Programação III

Guia de Referência: JAVA vs C vs PASCAL

Prof. Pedro A.B. Oliveira

1 Características Gerais

Comparativo estrutural entre a linguagem nova (Java), a que os alunos conhecem (C) e a referência estruturada clássica (Pascal).

Característica	JAVA (O Novo)	C (Base)	PASCAL (Ref.)
Sensibilidade	Case-Sensitive (Diferencia maiúsculas)	Case-Sensitive	Not Case-Sensitive
Paradigma	Orientado a Objetos (100% Classes)	Estruturado / Procedural	Estruturado / Procedural
Início (Main)	public static void...	int main()	program Nome;
Blocos	Chaves { ... }	Chaves { ... }	begin ... end
Declaração	Tipada (Exige declarar)	Tipada	Tipada
Memória	Automática (Coletor de Lixo)	Manual (malloc/free)	Manual (New/Dispose)

2 Variáveis e Tipos de Dados

Em Java, você pode declarar e inicializar na mesma linha. As variáveis devem ser declaradas antes do uso.

Tipo	Sintaxe JAVA	Equivalente PASCAL
Inteiro	int idade = 25;	var idade: Integer;
Real	double salario = 10.5;	var salario: Real;
Caractere	char letra = 'A';	var letra: Char;
Texto	String nome = "Pedro";	var nome: String;
Booleano	boolean ativo = true;	var ativo: Boolean;

Nota Importante: String em Java começa com letra Maiúscula porque é uma Classe (Objeto), não um tipo primitivo como int.

3 Entrada e Saída (Console)

- **Escrever:** Em C usa-se printf. Em Java usa-se `System.out.println`.

- **Ler:** Em C usa-se `scanf`. Em Java usa-se a classe `Scanner`.

Listing 1: Exemplo de Leitura e Escrita em JAVA

```
import java.util.Scanner; // Importar biblioteca

public class ExemploIO {
    public static void main(String[] args) {
        Scanner leitor = new Scanner(System.in);

        System.out.print("Digite sua idade: "); // Print sem pular linha
        int idade = leitor.nextInt();           // Ler inteiro

        System.out.println("Sua idade e: " + idade); // Print pulando linha
    }
}
```

4 Estruturas Condicionais

4.1 IF / ELSE (Se / Senão)

A lógica é idêntica ao C. Diferente do Pascal, não existe a palavra `then`.

JAVA / C

```
if (nota >= 6.0) {  
    System.out.println("Aprovado");  
} else {  
    System.out.println("Reprovado");  
}
```

PASCAL (Referência)

```
if nota >= 6.0 then  
begin  
    WriteLn('Aprovado');  
end  
else  
begin  
    WriteLn('Reprovado');  
end;
```

4.2 SWITCH / CASE (Escolha)

Substituto do `case ... of` do Pascal. **Atenção:** Em Java e C, o comando `break` é obrigatório para parar a execução.

```
int opcao = 1;  
switch (opcao) {  
    case 1:  
        System.out.println("Menu Iniciar");  
        break; // Se esquecer, ele executa o case 2 tambem!  
    case 2:  
        System.out.println("Menu Sair");  
        break;  
    default:  
        System.out.println("Opcao Invalida");  
}
```

5 Laços de Repetição (Loops)

5.1 FOR (Para)

Estrutura: `for (inicio; condicao; incremento).`

```
// Java (Igual ao C)  
for (int i = 0; i < 10; i++) {  
    System.out.println("Contador: " + i);  
}  
// Pascal Equivalente: for i := 0 to 9 do ...
```

5.2 WHILE (Enquanto)

Executa enquanto a condição for verdadeira. Testa no início.

```
while (contador < 10) {  
    System.out.println(contador);  
    contador++; // Incremento manual necessario  
}
```

5.3 DO-WHILE (Repita)

Cuidado: A lógica é invertida em relação ao Pascal.

- **Pascal (Repeat-Until):** Repete ATÉ QUE seja verdade.
- **Java (Do-While):** Repete ENQUANTO for verdade.

```
do {  
    System.out.println("Rodando...");  
} while (ativo == true);
```

6 Dicas de Ouro (Erros Comuns)

1. Comparação de Strings (Texto):

- **Errado:** `if (nome == "Pedro")` → Isso compara endereços de memória!
- **Correto:** `if (nome.equals("Pedro"))` → Isso compara o conteúdo do texto.

2. Booleanos:

- Em C, `if(1)` funciona (1 é verdadeiro).
- Em Java, **NÃO** funciona. O Java exige `true` ou `false`. Use `if(x == 1)`.

3. Funções Soltas:

- Em Java não existem funções "soltas" no arquivo. Todo método deve pertencer a uma `class`.