

Linguagem java

Palavras Reservadas

Todas as palavras-chave de Java são em minúscula

abstract	do	implements	private	this
boolean	double	import	protected	throw
break	else	instanceof	public	throws
byte	extends	int	return	transient
case	false	interface	short	true
catch	final	long	static	try
char	finally	native	strictfp	void
class	float	new	super	volatile
continue	for	null	switch	while
default	if	package	synchronized	assert

Atribuição e Delimitador

Usamos o operador = para atribuições:

- Ex.:

```
int numero = 1;
```

Usamos o operador ; para delimitarmos os comandos:

- Ex.:

```
int numeroUm = 10;  
int numeroDois = 13;
```

Comentários

Em Java é possível comentar código de três formas diferentes:

- **// Comentários de uma linha**
- **/* comentários de várias linhas */**
- **/** comentários para documentação */**
É utilizado pela ferramenta javadoc.
Deve preceder a declaração de uma variável, um método ou uma classe. O documento é gerado no formato HTML.

Operadores

Operador	Descrição
=	Atribuição
!	Operador lógico de negação (NOT)
> , < , >= , <=	Operadores lógicos de desigualdade
== , !=	Operadores lógicos de igualdade e diferença
&& ,	E lógico (AND) , OU lógico (OR)
+, - , * , / , %	Adição, Subtração, Multiplicação, Divisão, Módulo
+= , -= , *= , /= , %=	Operação com atribuição. "x += 2"; é o mesmo que "x = x + 2";
++ , --	Incremento (+1), decremento (-1) "x++; " é o mesmo que "x = x+1"

Tipos de Dados

TIPO	DESCRIÇÃO
boolean	Pode assumir o valor true ou o valor false
char	Caractere em notação Unicode de 16 bits. Serve para a armazenagem de dados alfanuméricos. Também pode ser usado como um dado inteiro com valores na faixa entre 0 e 65535.
byte	Inteiro de 8 bits em notação de complemento de dois. Pode assumir valores entre $-2^7=-128$ e $2^7-1=127$.
short	Inteiro de 16 bits em notação de complemento de dois. Os valores possíveis cobrem a faixa de $-2^{15}=-32.768$ a $2^{15}-1=32.767$
int	Inteiro de 32 bits em notação de complemento de dois. Pode assumir valores entre $-2^{31}=2.147.483.648$ e $2^{31}-1=2.147.483.647$.
long	Inteiro de 64 bits em notação de complemento de dois. Pode assumir valores entre -2^{63} e $2^{63}-1$.
float	Representa números em notação de ponto flutuante normalizada em precisão simples de 32 bits em conformidade com a norma IEEE 754-1985. O menor valor positivo representável por esse tipo é $1.40239846e-46$ e o maior é $3.40282347e+38$
double	Representa números em notação de ponto flutuante normalizada em precisão dupla de 64 bits em conformidade com a norma IEEE 754-1985. O menor valor positivo representável é $4.94065645841246544e-324$ e o maior é $1.7976931348623157e+308$

Identificadores

- São nomes dados as variáveis, classes ou métodos.
- Podem iniciar com uma letra Unicode*, um cifrão (\$) ou um underscore (_).
- Diferencia maiúsculas de minúsculas e não tem um tamanho máximo.
- Exemplo
 - meuNome
 - Nome_usuario
 - _valor
 - \$contador
 - HelloWorld
 - @mail

//ilegal

Exemplo

Nome do arquivo deve ser igual ao da classe pública.
"HelloWorld.java"

```
public class HelloWorld
{
    public static void main (String []args)
    {
        System.out.println("Hello World!!!!");
    }
}
```

Toda **aplicação** é executada através do método **main**, que possui sempre esta declaração

Compilação

```
C:\> javac HelloWorld.java //Compilador
```

- Erros comuns de compilação:
 - **Javac: Command not found** – variável de ambiente PATH não definida corretamente.
 - **HelloWorld.java: 4: Public class HelloWorld must be defined in a file called “HelloWorld.java”** – Apenas uma classe pública em Java pode ser declarada em cada arquivo .java, e o nome deste arquivo deve ter o mesmo nome desta classe pública.

Declaração de Classes

```
<modificadores> class <nome> {  
    <declaração de atributos>  
    <declaração de construtores>  
    <declaração de métodos>  
}
```

public, private,
final abstract ...

Qualquer identificador
válido que já não esteja
sendo usando

Exemplo

```
public class Retangulo  
{  
    int base, altura;  
    public int calculaArea() {return base*altura};  
}
```

Declaração de Atributos

`<modificadores> <tipo> <nome> [=<valor inicial>];`

Modificadores
Public, Final, Static

Tipo
byte, short, int, char,
float, double...

Nome
Qualquer identificador
válido e já não esteja
sendo usando

Valor inicial
Pode ser atribuído no
momento da
declaração do atributo

Exemplos

```
public int x;
```

```
static double calc = 45.76;
```

```
String nome = "Rita Cadillac";
```

Declaração de Métodos

```
<modificadores> <tipo retornado> <nome> (<parâmetros>) {  
    <expressão>  
}
```

Tipo retornado
void, byte, short, int,
char, float, double...

Parâmetros
Lista o número e tipo
dos argumentos que o
método pode receber,
separados por vírgula

Exemplo

```
public int calculaArea (int b, int a) {  
    int base = b, altura = a;  
    return base * altura;  
}
```



É hora de recapitular