



# Módulo 7





A decorative pattern of hexagons in various shades of blue and cyan on the left side of the slide. Some hexagons contain icons: a lightbulb, a thumbs up, a smartphone, a magnifying glass, and a gear. A large cyan hexagon in the center of this pattern contains the number 3.

# 3

# Classes, métodos modificadores e palavras reservadas

Entendendo modificadores de acesso



# Nomenclaturas


- Classes e interfaces: A primeira letra deve ser maiúsculas, caso o nome seja formado por mais de uma palavra, as demais palavras devem ter sua primeira letra maiúscula. (camelCase).
- Métodos: A primeira letra do método deve ser minúscula e após devemos aplicar o camelCase.
- Variáveis: Da mesma forma que os métodos.
- Constantes: Todas as letras devem ser maiúsculas.





# Nomenclaturas

Classes	Métodos	Variáveis	Constantes
Carro	desligar	motor	COMBUSTIVEL
CursoJava	iniciarTeste	quantMarcha	NOME_CURSO
Hotel	reservaSuite	nomePeca	TAXA_SERVICO

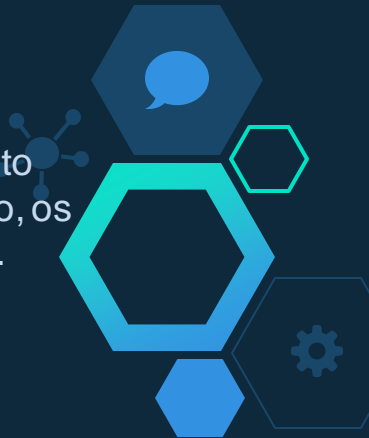




# Modificadores de acesso


Os modificadores de acesso são padrões de visibilidade de acessos às classes, atributos e métodos. Estes modificadores são palavras-chaves reservadas pelo [Java](#).

Por exemplo, quando um programa cria (instancia) um objeto da classe Banco, a variável senha é encapsulada (ocultada) no objeto onde pode ser acessada apenas por métodos da classe do objeto, os métodos getters e setters, que manipulam a variável de instância.





# Modificadores de acesso

- **private:** O modificador de acesso private quando aplicado a um atributo ou a um método indica que os mesmos só podem ser acessados de dentro da classe que os criou.
  - **protected:** A instrução protected indica que o método ou a variável assim declarada possa ser acessada somente dentro do pacote em que está contida através de uma subclasse.
- 



# Modificadores de acesso

- **public:** A instrução public indica que a classe, método ou variável assim declarada possa ser acessada em qualquer lugar, a qualquer momento da execução do programa.
- **Default (Padrão):** A classe e seus membros são acessíveis somente por classes do mesmo pacote.








# Modificadores de acesso

	<b>private</b>	<b>default</b>	<b>protected</b>	<b>public</b>
mesma classe	sim	sim	sim	sim
mesmo pacote	não	sim	sim	sim
pacotes diferentes (subclasses)	não	não	sim	sim
pacotes diferentes (sem subclasses)	não	não	não	sim



A series of hexagonal icons in various shades of blue and cyan are arranged along the left edge of the slide. The icons include a lightbulb, a thumbs-up, a network of nodes, a smartphone, a magnifying glass, a gear, and a speech bubble.

4

# Palavras reservadas

Entendendo palavras reservadas



# Palavras reservadas

<u>abstract</u>	<u>continue</u>	<u>for</u>	<u>new</u>	<u>switch</u>
<u>assert</u> (3)	<u>default</u>	<u>goto</u> (1)	<u>package</u>	<u>synchronized</u>
<u>boolean</u>	<u>do</u>	<u>if</u>	<u>private</u>	<u>this</u>
<u>break</u>	<u>double</u>	<u>implements</u>	<u>protected</u>	<u>throw</u>
<u>byte</u>	<u>else</u>	<u>import</u>	<u>public</u>	<u>throws</u>
<u>case</u>	<u>enum</u> (4)	<u>instanceof</u>	<u>return</u>	<u>transient</u>
<u>catch</u>	<u>extends</u>	<u>int</u>	<u>short</u>	<u>try</u>
<u>char</u>	<u>final</u>	<u>interface</u>	<u>static</u>	<u>void</u>
<u>class</u>	<u>finally</u>	<u>long</u>	<u>strictfp</u> (2)	<u>volatile</u>
<u>const</u> (1)	<u>float</u>	<u>native</u>	<u>super</u>	<u>while</u>



A decorative graphic on the left side of the slide. It features a large central hexagon with a blue-to-cyan gradient, containing the number '5'. Surrounding this central hexagon are several smaller hexagons of varying shades of blue and cyan. Some of these smaller hexagons contain white icons: a lightbulb, a thumbs-up, a smartphone, a magnifying glass, and a gear. There is also a network-like icon with a central node and radiating lines, and a speech bubble icon. The entire graphic is set against a dark blue background.

5

# Comentários



# Comentários

## Tipos de comentários

// este comenta somente uma linha

/\*

\* Este comenta várias linhas de uma só vez

\*/

/\*\*

\* Este comentário é o javadoc, serve para criar a documentação  
\* automática do sistema.

\*\*/



# Comentários

## Tags javadoc

Tag	Significado
<code>@author</code>	Especifica o autor da classe ou do método em questão.
<code>@deprecated</code>	Identifica classes ou métodos obsoletos. É interessante informar nessa tag, quais métodos ou classes podem ser usadas como alternativa ao método obsoleto.
<code>@link</code>	Possibilita a definição de um link para um outro documento local ou remoto através de um URL.
<code>@param</code>	Mostra um parâmetro que será passado a um método.
<code>@return</code>	Mostra qual o tipo de retorno de um método.
<code>@see</code>	Possibilita a definição referências de classes ou métodos, que podem ser consultadas para melhor compreender idéia daquilo que está sendo comentada.
<code>@since</code>	Indica desde quando uma classe ou métodos foi adicionado na aplicação.
<code>@throws</code>	Indica os tipos de exceções que podem ser lançadas por um método.
<code>@version</code>	Informa a versão da classe.



# Referências

- Templates para apresentações:
  - ◆ [https://www.slidescarnival.com/?utm\\_source=template](https://www.slidescarnival.com/?utm_source=template)
- Figura 1:
  - ◆ <https://docente.ifrn.edu.br/pedrobaesse/disciplinas/programacao-orientada-a-objetos/material-de-aula/aula-02-introducao-a-programacao-orientada-a-objeto>
- Figura 2:
  - ◆ <https://gastronominho.com/gordicas-o-que-se-como-no-cafe-da-manha-ao-redor-do-mundo/>

