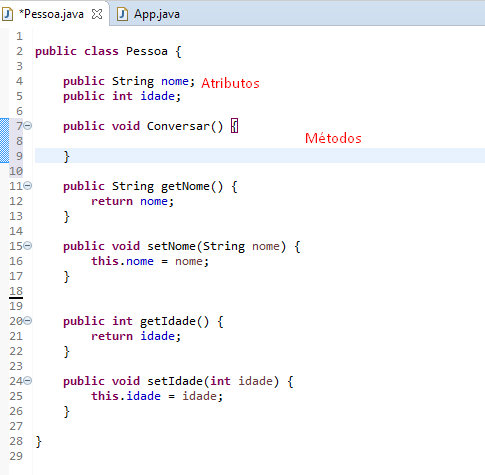
**Diferença entre Classe, Objeto, Atributo, Método e Construtor**

É comum em linguagem de programação ouvir falar sobre Orientação a objetos, o famoso ‘POO - Programação Orientada a Objetos’.

Porém, algumas das principais diferenças entre os componentes que permitem isso, não estao bem claro muitas vezes. Abaixo, uma pequena descrição do conceito de cada um.

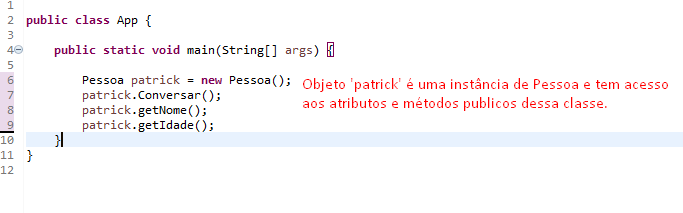
**Classe** - em programação, uma **classe** é um **tipo** definido pelo usuário - uma espécie de ‘projeto’ de um objeto -, algo que tenha características e comportamentos, ou seja, que possa guardar atributos e métodos para serem utilizados por uma instância de um objeto dessa classe. Uma classe normalmente é uma abstração de algo real, como por exemplo, uma pessoa, lugar, etc.

Uma classe possui um nome, visibilidade (public, private, etc) e suas características e ações. Abaixo, um exemplo:

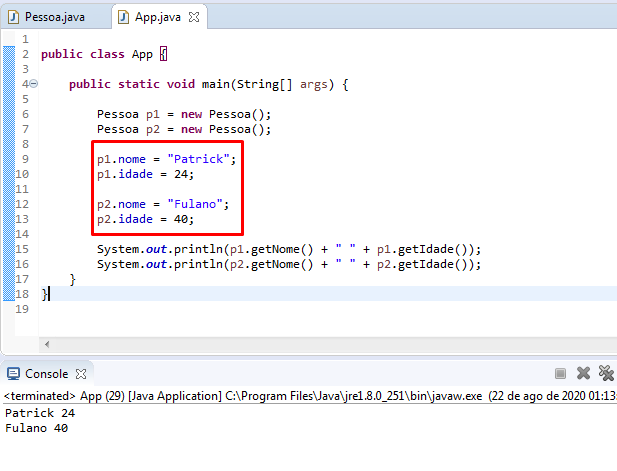


**Objeto** - são características definidas pela classe - uma instância de uma classe - utilizado para acessar atributos e invocar métodos públicos dessa classe.

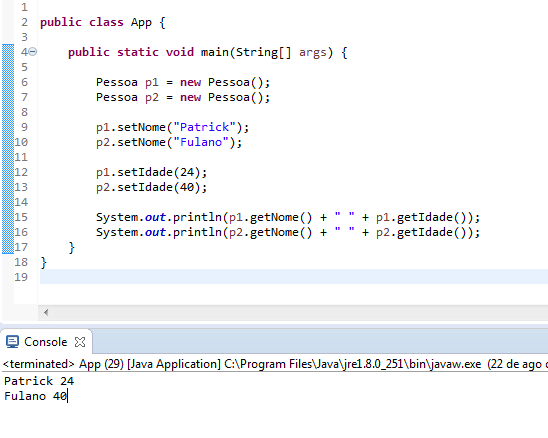
Normalmente instanciamos um objeto de uma classe em alguma outra classe para acessar informações da classe da qual esse objeto foi instanciado, como os atributos ou métodos.



**Atributos** - também conhecidos como variáveis são as propriedades de um objeto. Essas propriedades definem o estado de um objeto, fazendo com que esses valores possam sofrer alterações.

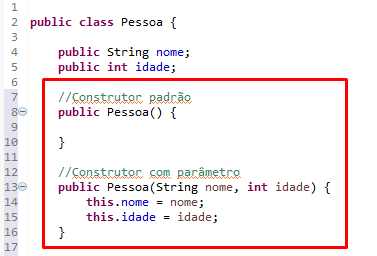


--

Poderia utilizar também o set() para setar o nome e a idade, a impressão ficaria a mesma:

**Métodos** - são as ações / procedimentos declarados na classe que podem ser invocados pelo objeto. No exemplo anterior, invocamos alguns métodos com o objeto instanciado da classe, como getNome() e getIdade().

**Construtor** - é um método especial que inicializa os atributos de um objeto quando ele é instanciado. Posso declarar um construtor sem parâmetro (padrão) ou com parâmetro, ou se preferir, os dois. A declaração de um construtor em uma classe é sempre feita utilizando o mesmo nome da classe, assim:



**Importante**: mesmo não declarado em código, toda classe tem um construtor padrão fornecido pelo compilador, para evitar possíveis erros de sintaxe. Esse construtor existe para possibilitar a criação de objetos para uma classe.