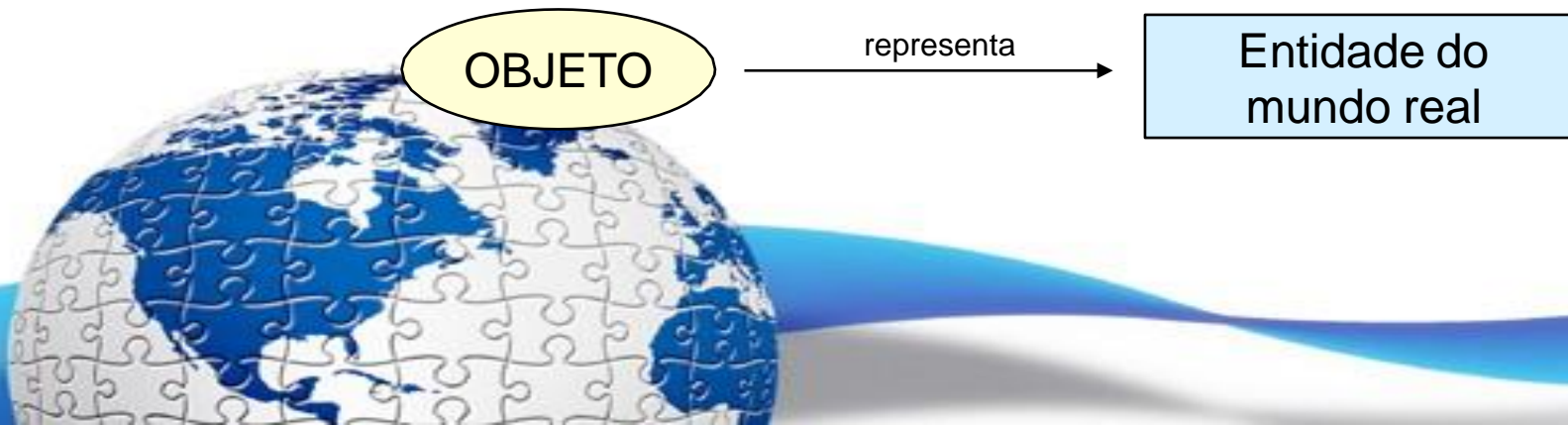


# Orientação a objetos

- Programação orientada a objetos!

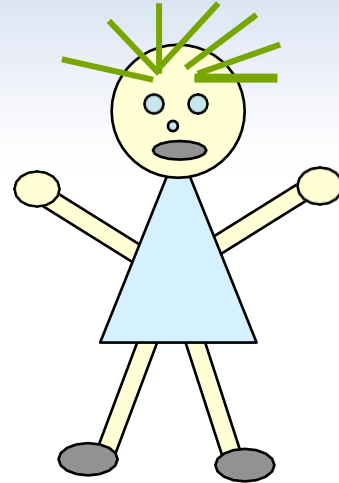
└→ Os problemas são pensados em termos de  
OBJETOS, suas propriedades e métodos

- Para um sistema de uma biblioteca, por exemplo, teremos a divisão de objetos, tais como Livro, Andar, Matéria, Atendente, Usuário, etc...



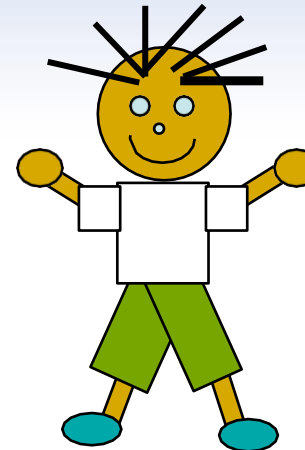
# Orientação a objetos

Floribela



- Nome: Floribela
- Sexo: feminino
- Cor do cabelo: verde
- Cor da roupa: azul
- Cor da pele: amarela
- Cor dos sapatos: vermelho
- Altura: 6cm
- Humor: assustada

Antoniolo



- Nome: Antoniolo
- Sexo: masculino
- Cor do cabelo: preto
- Cor da roupa: verde e branca
- Cor da pele: marrom
- Cor dos sapatos: azul
- Altura: 5,5cm
- Humor: feliz

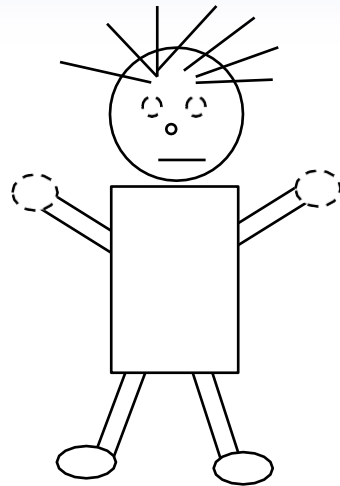
# Orientação a objetos

- Uma classe, então, vai representar o conjunto de objetos que possuem determinadas características em comum
- Ao definir uma classe, então, devemos definir dois pontos principais:
  - 1 – **atributos**, que são informações da classe (cor do cabelo, sexo, altura, etc...)
  - 2 – **métodos**, que são as ações que podem ser realizadas pelos objetos de cada classe (andar, correr, falar, pensar, etc...)

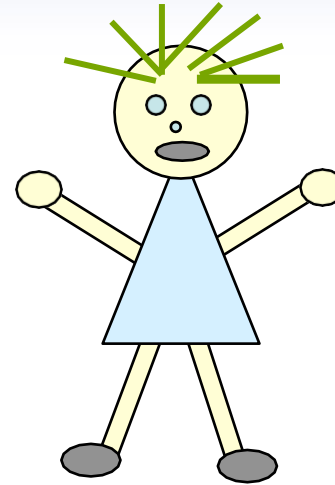


# Orientação a objetos

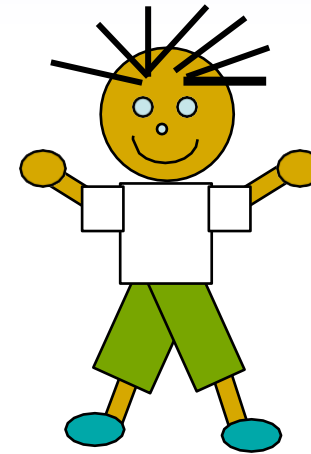
Classe Pessoa



Objeto Floribela



Objeto Antoniollo



Floribela e Antoniollo  
são **instâncias** da classe Pessoa



# Como criar classes no Java?

```
public class NomeDaClasse {
```

- ATRIBUTOS DA CLASSE

MÉTODOS DA CLASSE

}





# Exemplos de classes

public class Quero {  
    ~~Quero~~  
}



# Como criar atributos das classes no Java?

```
public class NomeDaClasse{  
    VISIBILIDADE TIPO NOME;  
}
```

Diagram illustrating the components of a Java class declaration:

- public**: Access modifier (circled)
- class**: Keyword (underlined)
- NomeDaClasse**: Class name (underlined)
- VISIBILIDADE**: Attribute visibility (circled)
- TIPO**: Attribute type (circled)
- NOME**: Attribute name (circled)



# Como criar atributos das classes no Java?

```
public class Pessoa{  
    private String nome;  
    private int idade;  
    private double altura;  
    private char sexo;  
}
```





# Instanciando objetos

TESTES

LIVRO

new Pessoa ("Geonito")

new Livro ();

new Pessoa?

TESTES



# Método construtor

```
public Pessoa (String nome) {  
    this.nome = nome;  
}
```

nome

~~nome = "guanto"~~



# Sobrecarga de métodos



# Métodos de acesso

GET

SET

```
public void setNome(String nome) {  
    this.nome = nome;  
}
```

