

# Introdução à Programação Orientada a Objetos

## Aula 2

# Orientação a Objetos

- Paradigma de análise projeto e programação
- Baseado na composição e interação entre diversas unidades de software chamadas de objetos
- Na programação orientada a objetos, implementa-se um conjunto de classes que definem os objetos presentes no sistema de software

# Histórico

- **Linguagem Simula 67**
  - Aplicação de idéias de POO
  - Criada em 1967
  - Por Ole Dahl e Kirsten Nygaard
- **Linguagem Smaltalk**
  - Totalmente OO
  - Por Alan Kay
  - Mais informações em <http://www.smaltalk.org>

# Vantagens

- Encapsulamento que reduz bugs
- Extensão que permite reaproveitamento de código

# Desvantagens

- **Aprendizado mais difícil**
- **Desempenho menor**

# Java? Por que?

- É obrigatoriamente orientada a objetos
- É simples: nos elementos e estruturas de programação
- É portátil: código-fonte pode ser compilado e vários SO's
- É gratuita: máquina virtual e software de desenvolvimento
- É robusta: Administração de memória mais segura
- Tem bibliotecas prontas: Acesso a BD's, GUI's, Internet

# O kit de desenvolvimento

- **JSDK SE**
- **Video do Download**
- **Video da instalação**

# Ambiente Interativo

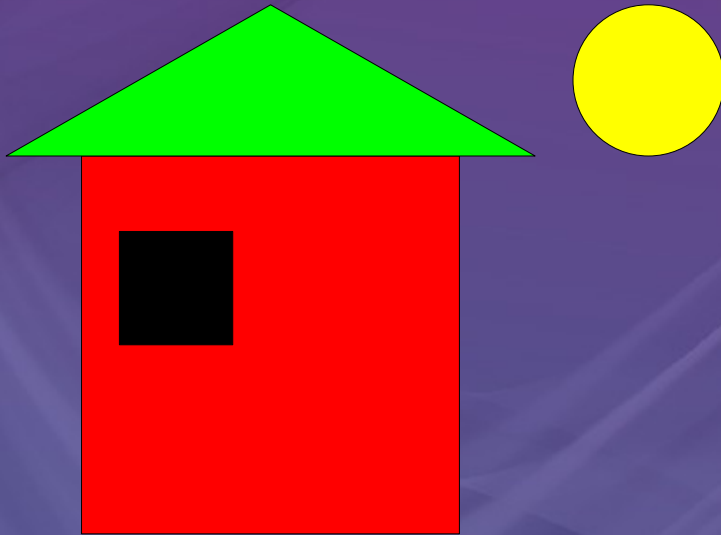
- BlueJ
- Video do Download
- Video da instalação

# Exemplo do Shapes

- Abrir projeto exemplo do Shapes
- Compilar o projeto
- Instanciar um circulo (new)
- Torná-lo visível (setVisible)
- Mover para os lados (moveUp, moveDown, moveLeft...)
- Mover em pixels na vertical e horizontal (moveVertical(int))
- Mudar o tamanho (changeSize(int))
- Mudar a Cor (String""")

# Exercício do Shapes

- Construir a casinha

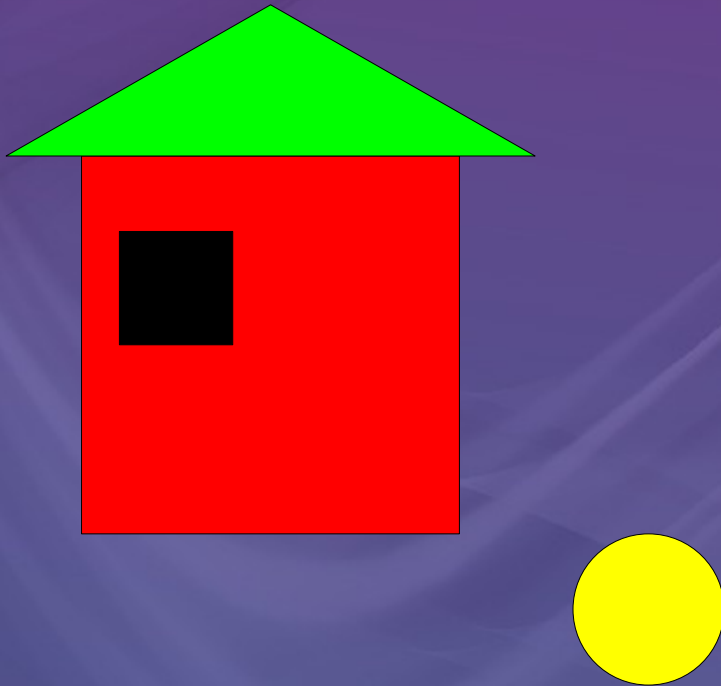


# Exemplo do Picture

- Abrir projeto exemplo do Picture
- Compilar o projeto
- Instanciar um picture (new)
- Desenhá-lo (draw)

# Exercício do Picture

- Faça o sol se por lentamente.



# Exemplo da Carteira

- 1. Criar um programa em Java OO que permita:
  - a. Criar uma nova carteira
  - b. Iniciar seu saldo com 0
  - c. Mostrar seu saldo
  - d. Acrescentar um Real ao saldo
  - e. Retirar uma quantidade qualquer do saldo

# Solução

- 1. Clique em Project/New Project e informe: ProjCarteira
  - a. Clique no botão New Class, informe o nome Carteira e clique em Class.
  - b. Abra o editor da classe, apague o conteúdo da classe e declare `int saldo;`
  - c. Implemente

```
int mostrarSaldo()  
{  
    return saldo;  
}
```

- - **d. Implemente**

```
void acrescentarUmReal()  
{  
    saldo = saldo + 1;  
}
```

- **e. Implemente**

```
void retirarQuantidade(int quanto)  
{  
    saldo = saldo - quanto;  
}
```

# Exercício da Caixa D'água

- 1. Criar um programa em Java OO que permita:
  - a. Criar uma nova caixaD'água
  - b. Iniciar seu nível com 0
  - c. Mostrar seu nível
  - d. Acrescentar uma quantidade qualquer de água
  - e. Esvaziar um Metro de agua