

**INSTITUTO
FEDERAL**

Piauí

Campus
Pedro II



Programação orientada a objetos

Aula 01: Introdução à
Programação Orientada a Objetos

Objetivos

- Conhecer o paradigma orientado a objetos;
- Analisar os principais conceitos do paradigma;
- Comparar OO com programação estruturada;

Introdução

Paradigma de programação que facilita a escrita e manutenção dos códigos, adotando conceitos e técnicas que abstraem características do mundo real;

Criado por Alan Kay (criador da linguagem Smalltalk);

- Alto índice de abstração

- Fácil entendimento

- Manutenção

- Reaproveitamento

- Produtividade

Paradigmas



Paradigmas

Consiste no mapeamento do problema no mundo real a ser resolvido num modelo computacional.

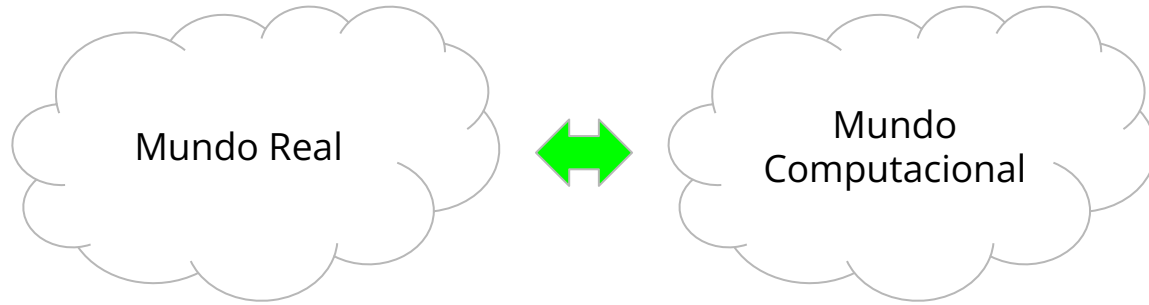
- **Programação Estruturada** – Consiste na criação de um conjunto de procedimentos (algoritmos) para resolver o problema – Encontrar modos apropriados de armazenar os dados
- **Programação Orientada a Objetos** – Consistem em identificar os objetos e as operações relevantes no mundo real – O mapeamento desses em representações abstratas no espaço de soluções

Programação Orientada a Objetos

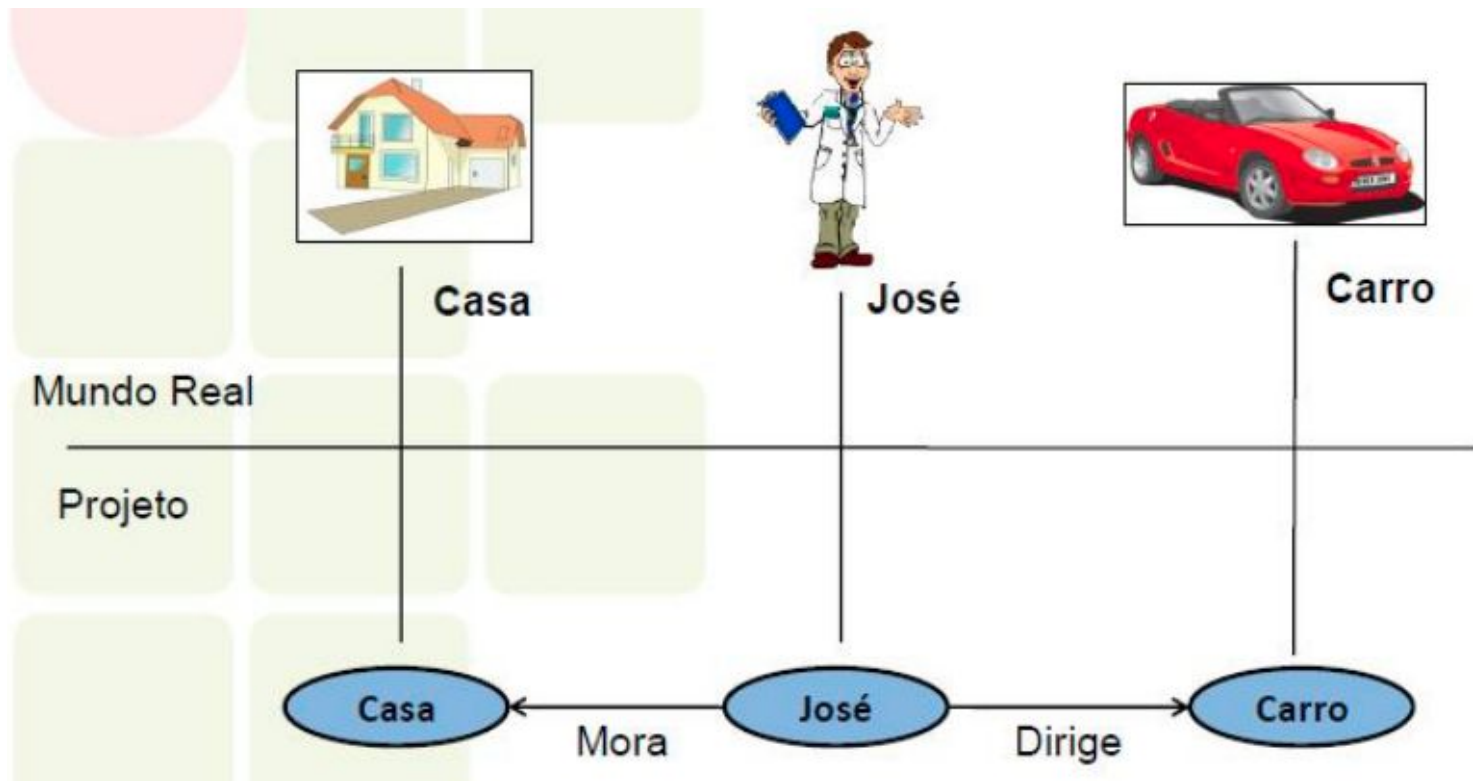
Paradigma de Programação – **Dominante nos dias atuais**

- Substituiu as técnicas de programação procedimental (estruturada)
- “Fornece um mapeamento direto entre o mundo real e as unidades de organização utilizadas no projeto”
- Diversas unidades de software, chamadas de objetos, que interagem entre si
- Separa claramente a noção de o que é feito de como é feito

Programação Orientada a Objetos



Programação Orientada a Objetos



Programação Orientada a Objetos



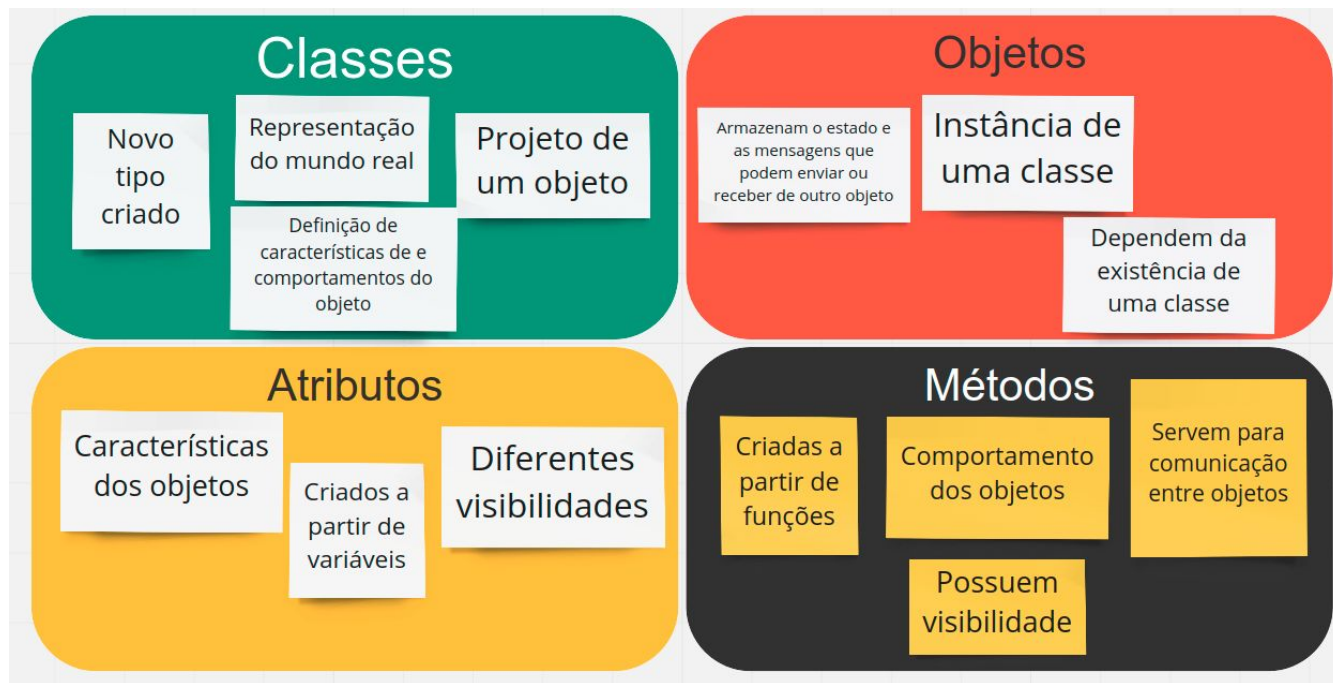
Programação Orientada a Objetos

Se baseia em **quatro (4) pilares básicos** para o desenvolvimento de aplicações completas;



Programação Orientada a Objetos

Conceitos essenciais



Exemplo

Construa um programa que simule uma conta corrente.

O dono desta conta deve se chamar João.

O saldo inicial desta conta deve ser R\$1000,00.

A conta deve estar vinculada ao cliente “João”, que possui CPF de 345.654.777-10.

Faça as seguintes operações nesta conta:

- Depósito de R\$1000,00

- Saque de R\$150,00

- Transferência de R\$200,00 para a conta de um amigo (nome: “José”, CPF: “009.223.443-90”).

Número das contas (João : 123-4 José: 567-0)

Considere que José tinha uma conta com saldo de R\$ -400,00.

Solução Estruturada

python

```
# Programa estruturado - controle de conta bancária
```

```
# Dados representados por variáveis e dicionários
```

```
conta = {"titular": "João", "saldo": 1000}
```

```
def depositar(conta, valor):  
    conta["saldo"] += valor
```

```
def sacar(conta, valor):  
    if conta["saldo"] >= valor:  
        conta["saldo"] -= valor  
    else:  
        print("Saldo insuficiente")
```

```
def exibir_saldo(conta):  
    print(f"Titular: {conta['titular']} - Saldo: R${conta['saldo']:.2f}")
```

```
# Programa principal
```

```
depositar(conta, 200)
```

```
sacar(conta, 150)
```

```
exibir_saldo(conta)
```

Solução 0.0.

python

Programa orientado a objetos - controle de conta bancária

class ContaBancaria:

def __init__(self, titular, saldo_inicial=0):

 self.titular = titular

 self.saldo = saldo_inicial

def depositar(self, valor):

 self.saldo += valor

def sacar(self, valor):

if self.saldo >= valor:

 self.saldo -= valor

else:

print("Saldo insuficiente")

def exibir_saldo(self):

print(f"Titular: {self.titular} - Saldo: R\${self.saldo:.2f}")

Programa principal

conta1 = ContaBancaria("João", 1000)

conta1.depositar(200)

conta1.sacar(150)

conta1.exibir_saldo()

Comparativo

Paradigma Estruturado

depositar(conta, valor)



sacar(conta, valor)



exibir_saldo(conta)

Titular: João
Saldo: R\$1000,00

Dados separados

Paradigma Orientado a Objetos

```
class ContaBancaria
  depositar(valor)
  sacar(valor)
  exibir_saldo()
```



Titular: João
Saldo: R\$1000,00

Dados +
comportamento

Próxima aula . . .

Primeiro pilar da Orientação a Objetos

Abstração