



**INSTITUTO FEDERAL DE  
EDUCAÇÃO, CIÊNCIA E TECNOLOGIA  
SERTÃO PERNAMBUCANO  
Campus Salgueiro**

**Sistemas Para Internet – 2021.1**

**Programação Orientada a Objetos**

**Atividade 01**

### **Linguagens de Programação**

- Criadas para facilitar a comunicação entre computadores e humanos, mas também entre humanos com humanos no que diz respeito à programação das máquinas.
- Possui regras de sintaxe e semânticas criadas artificialmente para facilitar o entendimento e assimilação dos seres humanos.

#### **Alguns paradigmas de Programação**

- 1. Programação Imperativa**
- 2. Programação Orientada a Objetos**
- 3. Programação Funcional**
- 4. Programação Lógica**

Os paradigmas apresentam formas diferentes de pensar na lógica por trás da resolução dos problemas.

#### **Programação Imperativa**

Este paradigma, como o nome já diz, está relacionado a definir comandos a serem seguidos pela máquina para serem executados.

#### **Programação Orientada a Objetos**

O programa trabalha com a ideia de objetos, que se relacionam e interagem entre si, transformando seus estados.

#### **Programação Funcional**

Paradigma funcional trata a resolução dos problemas computacionais através de funções matemáticas, com entradas, processamento e saída.

#### **Programação Lógica**

Os problemas são resolvidos na declaração do resultado que o programa deve ter, ao invés de como será resolvido. As declarações criam um conjunto de todas as soluções possíveis para o tipo de problema.

### **História das Linguagens de Programação**

As primeiras linguagens desenvolvidas foram as linguagens de máquina. Linguagens essas que se tratava apenas de números binários, que é a forma como as máquinas realizam suas funções. Para o ser humano, é praticamente impossível de se trabalhar usando apenas linguagem de máquina. Com isso foi desenvolvido o Assembly na década de 1940. Já na década de 1950, as linguagens de alto nível foram surgindo. Estas mais parecidas com as

linguagens utilizadas pelos humanos, e assim mais fáceis de entender e se desenvolver programas e instruções para os computadores.

Os primeiros programas desenvolvidos na década de 1940 foram para calcular as trajetórias de balas durante a Segunda Guerra Mundial. Nasceu então as linguagens Fortran I, Algol, APL.

Após a guerra, os computadores foram utilizados pelas empresas para desenvolver aplicações úteis nos trabalhos diários. Melhorando a precisão, agilidade e confiabilidade dos dados. Por exemplo, sistemas de contabilidade, folhas de pagamentos etc. Assim o COBOL foi criado e inclusive é uma linguagem que ainda nos dias de hoje é utilizada em sistemas legado.

Na década de 1970, muitas linguagens foram desenvolvidas para serem utilizadas no ensino de programação. Devido a cada vez mais necessidade de profissionais na área, foi preciso também desenvolver uma forma de ensinar. Linguagens dessa época: BASIC, Pascal, Modula.

No final dos anos 70, foram surgindo a necessidade de desenvolver programas maiores e robustos, como por exemplo Sistemas Operacionais e Máquinas Virtuais. Neste período surgiu a Linguagem C.

Já nos anos 90, a grande explosão da internet surgiu. Criando assim linguagens voltadas para a web, com o foco na interação com o usuário, assim foram desenvolvidas as linguagens Perl, PHP, Python, Java etc.

Na era atual estamos lidando com Inteligência Artificial, com o foco na otimização de recursos, com isso se utiliza as linguagens já existentes (Python, C++, Java, LISP e Prolog) e que com o passar dos anos foram melhoradas para que o uso da IA fosse possível.

## JAVA

Java é uma linguagem que está presente no desenvolvimento de aplicações móveis, *web apps*, *big data*, *machine Learning*, *IoT*, sistemas embarcados, e várias outras tecnologias. Tem como filosofia o WORA: *Write Once, Run Anywhere* (Escreva uma vez, e rode em qualquer lugar - tradução livre).

Criada em 1995 pela Sun Microsystems e consequentemente adquirida pela Oracle em 2008. Por ser uma linguagem multiparadigma, ela apresenta vários diferentes paradigmas de programação, mas mais comumente relacionada a Orientação a Objetos.

O código Java não é compilado para o código nativo. Ele é compilado para *Bytecode* que será interpretado pela *Java Virtual Machine* (JVM). É graças a esta funcionalidade que o Java pode ser transportado para qualquer tipo de plataforma. Uma funcionalidade muito interessante do Java é que mesmo códigos escritos em versões anteriores continuam sendo executados nas versões mais novas.

Por sua versatilidade, o Java tem versões específicas para desenvolvimento, grandes corporações e apenas utilização pelos usuários.

Java é conhecida por ser bastante verborrágica, por conta de sua sintaxe. Mas isso tem melhorado com as suas atualizações. Também é *Camel Sensitive*, onde letras maiúsculas e minúsculas tem suas importâncias e formas de uso.

## Programação Orientada a Objetos

É um paradigma de desenvolvimento que várias linguagens de programação utilizam e que vai de encontro com o paradigma procedural/estruturado.

P00 não é apenas um paradigma de programação, ele também está presente nas análises, levantamento e de projeto. Sendo assim necessário entender todo o conceito para que se possa ser aplicado não só no desenvolvimento do código em si.

A motivação do P00 é representar cada elemento do mundo real em forma de objeto. Dessa forma, o projeto deve ser relacionado a um objeto no mundo real, ou seja, sendo um conjunto de estados e comportamentos.

### Pilares da Orientação a Objetos:

- Herança
  - Capacidade de um objeto ser idealizado baseado em outro objeto. Objetos pais que contém atributos e métodos, eles podem ser estendidos para um novo objeto filho. Podendo assim replicar as características de um objeto para outro.
- Polimorfismo
  - É a capacidade de um objeto que, mesmo tendo herdado características e métodos de um objeto pai, possa ser reescrito para realizar algo totalmente diferente do que seu objeto pai faz.
- Encapsulamento
  - Capacidade de esconder detalhes da implementação do objeto, expondo apenas aquilo que pode ser acessado publicamente. Sendo possível programar adicionando uma camada de segurança à aplicação.
- Abstração
  - Representar um objeto de uma forma abstrata, obrigatoriamente sendo herdado por outras classes. Podendo criar uma classe abstrata com atributos e métodos mas a implementação deve ser feita pelas classes filhas.