

Programação Orientada a Objetos

Introdução

Alexandre Mello

Fatec Campinas

2024

Roteiro

1 Introdução

2 PE versus POO

3 História

4 Java

5 Python

Introdução

- Normalmente os comandos de um programa são executadas sequencialmente
- Em C (e C++) vários comandos permitem que essa sequencia seja quebrada, causando o que é chamado de transferência de controle
- Durante a década de 60 tornou-se claro que o uso indiscriminado de transferência de controle era a raiz de muitos problemas

Programação Estruturada

- Dijkstra (1968) provou que independente da complexidade envolvida, todo programa poderia ser escrito como uma combinação de comandos primitivos envolvendo três estruturas básicas de controle: sequencia, seleção, repetição
- Assim nascia a Programação Estruturada
- A experiência tem mostrado que a melhor forma de se desenvolver programas de grande porte é construí-los a partir de pequenas partes (dividir para conquistar) – funções

Programação Orientada a Objetos

- Devido os requisitos atuais, os softwares têm se tornado cada vez mais complexos e maiores
- Isso tem levado a busca de meios para tornar a tarefa de programação mais produtiva
- Ainda não existe uma resposta definitiva a essa busca, mas há um consenso de que a Programação Orientada a Objetos (POO) consegue produzir resultados mais competitivos do que as abordagens atualmente empregadas

Análise Estruturada versus Orientada a Objetos

- AE

- ▶ Procedimentos são implementados em blocos e a comunicação entre eles se dá pela passagem de dados
- ▶ Um programa estruturado, quando em execução, é caracterizado pelo acionamento de procedimentos cuja tarefa é a manipulação de dados

- AOO

- ▶ Dados e procedimentos são encapsulados em um só elemento denominado objeto
- ▶ O estabelecimento de comunicação entre objetos (envio e recebimento de mensagens) caracteriza a execução do programa

Vantagens da AOO

- Maior índice de reaproveitamento de código
- Maior facilidade de manutenção
- Menor código gerado
- Maior confiabilidade no código
- Maior facilidade de gerenciamento do código (reduz grandes problemas para menores)
- Maior robustez, etc.

História

- A ideia da Orientação a Objetos surgiu a partir das técnicas empregadas para o desenvolvimento de hardwares, onde pedaços simples de hardware (chips) eram unidos para se montar um hardware mais complexo
- Historicamente, a Orientação a Objetos foi consequência do amadurecimento de princípios já existentes na década de 60 (linguagem Simula, 1960-67), reimplementados na década de 70 (linguagem Smalltalk) e incorporados a novas linguagens ou linguagens ampliadas na década de 80 (linguagem C++)

Características

- Orientada a objetos: tudo são classes ou instâncias com exceção dos tipos primitivos
- Sem ponteiros: não permite a manipulação direta de endereços de memória
- Coletor de lixo: destroi e retira do heap (memória) os objetos não utilizados
- Facilita o uso de multithreading

Características

- Independente de plataforma: classes compiladas para forma intermediária (bytecode)
- Tratamento de exceções
- Máquina Virtual Java: responsável por interpretar os bytecodes
- Segurança: fortemente tipada e com assinaturas digitais para bytecode

Java SE

- JRE (JSE Runtime Environment): ambiente de execução já com JVM
- JDK (JSE Development Kit): kit de desenvolvimento

Fases de um programa em Java

- Fase 1 (Edição): Editar um arquivo com código em Java e salvá-lo com a extensão `.java`
- Fase 2 (Compilação): Comando `javac` compila o programa e gera bytecode. O compilador Java traduz `.java` em `.class`
- Fase 3 (Carga): Carrega o programa na memória antes de ser executado. Um carregador de classes (java), pega o arquivo `.class` que contém bytecode, e inicia a JVM
- Fase 4 (Verificação): O verificador assegura que o bytecode é válido e não viola as restrições de segurança de Java
- Fase 5 (Execução): A máquina virtual Java (Interpretador) interpreta em tempo de execução (JIT) o programa, e realiza as chamadas correspondentes à plataforma nativa

Instalação

- Acesse <http://java.oracle.com>, obtenha, sob o link da tecnologia Java SE, a versão mais nova do Java SE Development Kit (JDK) para o seu SO e instale-o
- Configurar as variáveis de ambiente do SO (este passo só é importante se for compilar e executar em linha de comando)
 - 1 PATH - deve incluir o caminho até o diretório bin, logo abaixo do diretório da instalação do JDK
 - 2 CLASSPATH - deve descrever todos os caminhos para locais onde serão armazenadas classes e pacotes Java
 - 3 JAVA_HOME - caminho até o diretório de instalação do JDK
- Obtenha e instale algum IDE para desenvolvimento Java

Python

Características

- Usa uma sintaxe elegante, tornando os programas que você escreve mais fáceis de ler
- É uma linguagem fácil de usar que torna simples fazer seu programa funcionar
- Vem com uma grande biblioteca padrão que suporta muitas tarefas de programação comuns
- O modo interativo do Python facilita o teste de pequenos trechos de código
- Multi-plataforma

Características

- Uma variedade de tipos de dados básicos estão disponíveis
- Python suporta programação orientada a objetos e herança múltipla
- A linguagem suporta captura de exceções, resultando em um tratamento de erros mais limpo
- Os tipos de dados são dinamicamente tipados
- Python contém recursos de programação avançados, como geradores e list comprehensions
- O gerenciamento automático de memória do Python evita que você tenha que alocar manualmente e liberar memória em seu código