

1) Dê 3 definições para Paradigma de Programação Orientada a Objetos.

A programação orientada a objetos surgiu como uma alternativa as características da programação estruturada, o sentido de sua criação foi o de aproximar a estrutura de um programa com o manuseio de coisas reais.

A programação orientada a objetos se baseia em conceitos de chave e objeto.

É um modelo de programação onde as classes possuem atributos que definem os objetos da vida real, as classes reúnem os objetos com características em comum, descrevendo seus serviços e informações.

2) Explique a diferença entre classe e objeto.

Um objeto possui características e seus comportamentos, como por exemplo um carro, onde cor, modelo são características e acelerar, buzinar são comportamentos. Continuando com o carro como exemplo, podemos observar que ao entrar em uma concessionária por exemplo temos vários carros com características similares ou diferentes, mas continuam sendo carros, a classe entra para classificar esse objeto (carro), sendo assim o objeto é uma instância da classe chamada carro e a classe é um conjunto de características e comportamentos que definem o conjunto de objetos pertencentes à essa classe.

3) O que são atributos?

Os atributos são as características do objeto, digamos que são as variáveis. No exemplo do carro, podemos ter os atributos cor, modelo, marca.

4) O que é a UML?

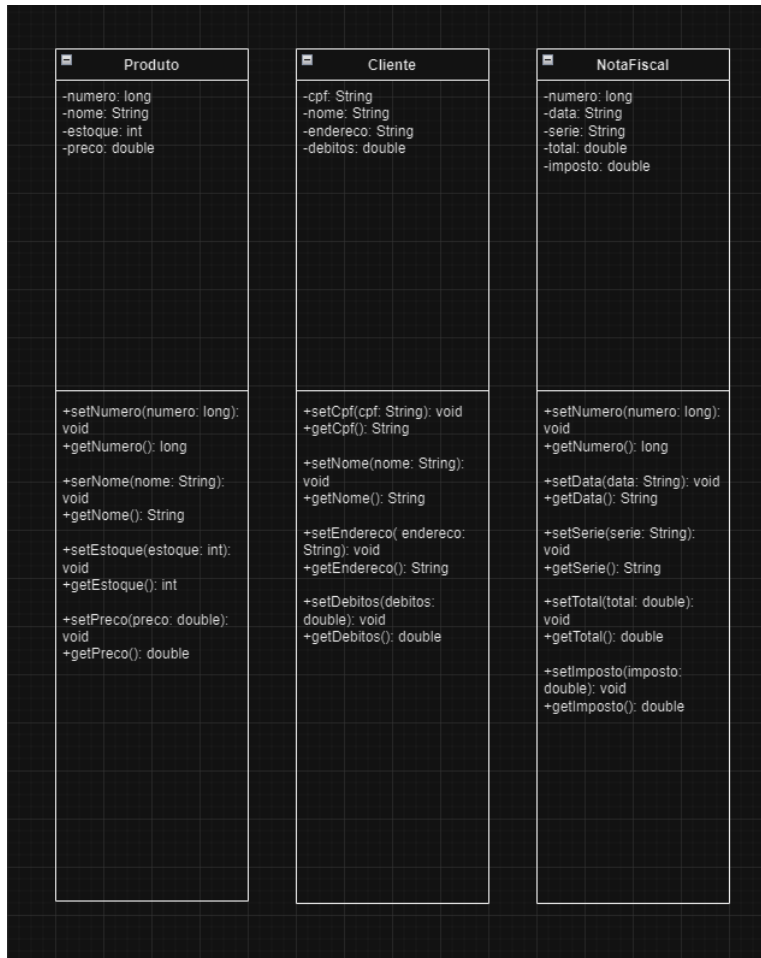
Unified Modeling Language, é uma linguagem que define uma série de artefatos que nos ajuda na tarefa de modelar e documentar os sistemas orientados a objetos.

5) O que é diagrama de classes da UML?

Ele documenta o que o sistema faz do ponto de vista do usuário. Em outras palavras, ele descreve as principais funcionalidades do sistema e a interação dessas funcionalidades com o usuário, mapeando de forma clara a estrutura de um determinado sistema ao modelar suas classes, atributos.

6) Como é representada uma classe na UML? Dê 3 exemplos.

As classes são representadas por um retângulo que inclui linhas do nome da classe, seus atributos e suas operações.



7) O que são modificadores de acesso public, private, protected? Dê exemplos.

São as definições de nível de acesso, onde o Public representa que o acesso é permitido a qualquer um, o Private representa que o acesso é restrito, privado, já o Protected tornar o membro acessível as classes do mesmo pacote através de herança.

8) O que são métodos?

Os métodos são as ações que os objetos podem exercer para interagir e se comunicar com outros objetos, no exemplo do carro, temos os métodos acelerar, buzinar, freiar.

9) O que são métodos getters e setters?

São os métodos que possuem acesso as variáveis privadas, são usados para proteger os dados, e acessa-los para determinado propósito. Os métodos getters são utilizados para retornar um valor, já os métodos setters são utilizados para definir ou atualizar um valor.

10) Em Java, o que é assinatura de um método?

Assinatura de um método é formada pelo seu nome e pelo tipo, quantidade e ordem de seus parâmetros. Desta forma, é possível existirem na mesma classe, métodos com o mesmo nome, desde que tenham listas de parâmetros diferentes.

11) Qual a convenção de identificadores (nomes) para classes, atributos, variáveis e métodos?

Classes: Os nomes de classe devem ser substantivos, em maiúscula e minúsculas com a primeira letra de cada palavra interna em maiúscula. Tente manter seus nomes de classe simples e descritivo. Sempre evite palavras-ligadas, evite todas siglas e abreviaturas, seja semântico.

Variáveis: Nos nomes de variáveis não devem começar com underscore _ ou sinal de dólar \$. Os nomes de variáveis devem ser curtos, mas significativos. A escolha de um nome de variável deve ser concebido para indicar ao observador casual a intenção da sua utilização. Nomes comuns para variáveis temporárias são l, j, k, m, n e para inteiros c, d, e.

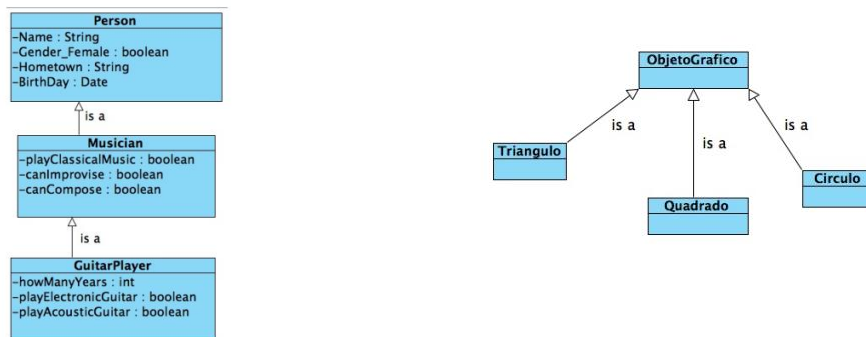
Métodos: Devem ser verbos, com a letra minúscula em primeiro lugar, com a primeira letra de cada palavra interna em maiúscula.

12) O que é herança?

Esse princípio diz que, uma classe pode compartilhar métodos e atributos entre si como, por exemplo, em um sistema escolar, onde temos uma classe Pessoa que contém características que definem uma pessoa. Porém, dentro do sistema temos uma outra classe Funcionário, que contém os mesmos atributos de Pessoa, além de outros atributos que apenas funcionários podem ter. Outro exemplo seria uma classe Aluno, que também contém atributos de pessoas e outros atributos que apenas pertencem a aluno. Outras classes que herdam de Pessoa recebem a palavra reservada "extends", que indica que as mesmas contêm as propriedades presentes na classe Pessoa.

13) Como a herança é representada na UML? Dê 2 exemplos

A Representação no diagrama UML da relação de Herança entre classe é feita por meio de uma linha ligando as duas classes e um símbolo triângulo apontando para Superclasse.



Fontes:

<http://www.professorvida.com.br/biblioteca/pdf/if62c/herancaJava.pdf>

<https://www.devmedia.com.br/convencoes-de-codigo-java/23871>

<https://www.lucidchart.com/pages/pt/o-que-e-diagrama-de-classe-uml>

<https://www.devmedia.com.br/o-que-e-uml-e-diagramas-de-caso-de-uso-introducao-pratica-a-uml/23408>

<http://fabrica.ms.senac.br/2015/03/classes-objetos-atributos-e-metodos-em-java/>

<https://www.devmedia.com.br/programacao-orientada-a-objetos-e-programacao-estruturada/32813>

<https://www.alura.com.br/artigos/poo-programacao-orientada-a-objetos>