|  |  |
| --- | --- |
|  | **AARE**: **Paradigmas de Linguagens de Programação**  **Professor**: Dr. Ausberto S. Castro V.  **Orientação a Objetos** |

**Prova JAVA**

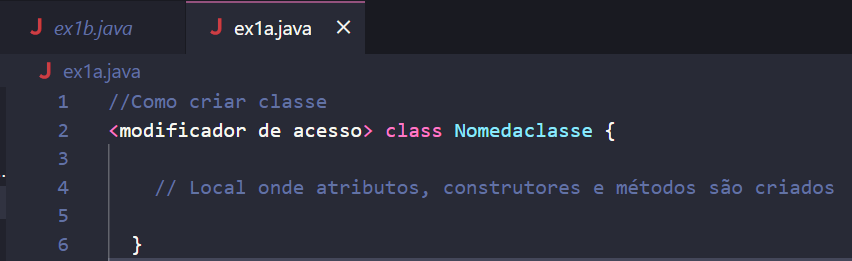
Nome Completo: Rômulo Souza Fernandes

Data: 19 de outubro de 2022

1. **(1 pontos) Responda detalhadamente as seguintes questões (exemplifique):**
   1. **Qual é a diferença entre uma classe e um objeto?**

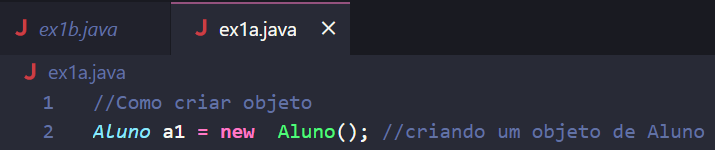
Na programação orientada a objetos uma classe possui determinados atributos, que são características. Esses atributos podem possuir um valor inicial, que é determinado por um construtor, na sua criação, o atributo então tem um valor padrão que é definido, além das ações realizadas sobre as classes ou pelas classes, chamamos isso de procedimentos. Contudo, entendemos que uma classe é um molde que informa os atributos que são as características e/ou os procedimentos que são as ações que a classe pode realizar.

Exemplo da criação de uma classe:



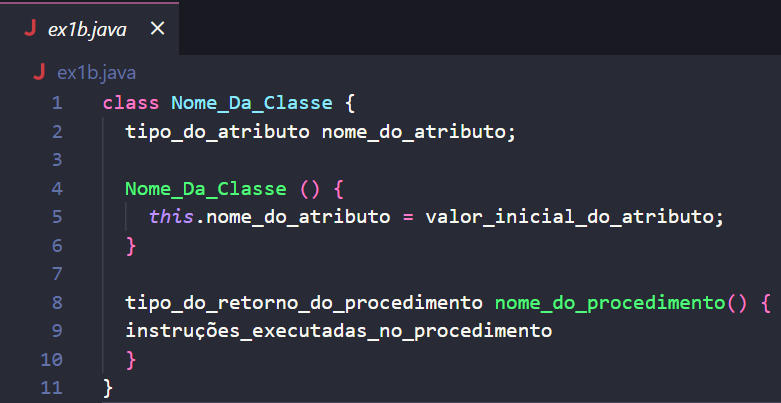
Na programação orientada a objetos um objeto é definido por uma classe e também é baseado em uma. Assim, quando um objeto é criado com base em uma certa classe, ele assume que possuirá os mesmos métodos e atributos que foram definidos pela classe. Podendo criar diversos objetos diferentes entre si que possuem o mesmo molde.

Exemplo da criação de um objeto:



* 1. **Explique como se define uma classe em Java**

Para se definir uma classe na linguagem Java é necessário:



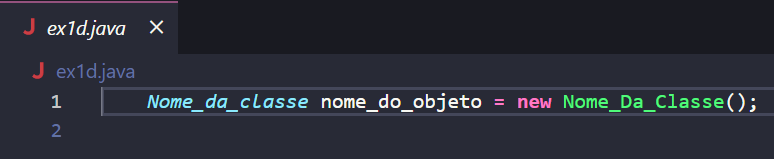
* 1. **Por que uma classe é uma abstração lógica?**

Para que uma classe seja criada é necessário compreender o seu funcionamento essencial e assim é possível definir amplamente, essa definição ampla é definido como abstração do objeto, objeto esse que temos como objetivo definir as ações e as características. A abstração necessita ser lógica e ter coerência, com isso uma classe é uma abstração lógica.

* 1. **Como os objetos são criados?**

Na POO os objetos são criados através da instancia de uma classe que anteriormente foi definida e abstraída, assim é possível criar objetos derivados da classe e cada um desses objetos criados através dessa classe é denominado como instância da classe.

Exemplo:



* 1. **O que é coleta de lixo e como ela funciona?**

O Garbage Collection ou coleta de lixo é usado em várias linguagens de programação, como Java e Python, para limpar a memória, atuando de forma a buscar objetos que não estão sendo usados, após achar esses objetos que são considerados lixo, eles são removidos da memória para que ela seja limpa.

1. **(1 pontos)** Desenhar todas as classes, subclasses e superclasses associadas aos seguintes objetos, indicando em cada caso, pelo menos 2 atributos e 2 métodos, e a relação de herança entre classes
   1. Carro
   2. Fruta
   3. Mamífero
   4. Computador
   5. Aparelho eletrônico
2. **(01 ponto)** Dar três exemplos de cada operação abstrata em orientação a objetos

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | **Conceito** | **Exemplo 1** | **Exemplo 2** | **Exemplo 3** |
| **Instanciação** | Classe → Objeto |  |  |  |
| **Classificação** | Objeto → Classe |  |  |  |
| **Especialização** | Classe geral →  Classe específica |  |  |  |
| **Generalização** | Classe específica →  Classe geral |  |  |  |
| **Decomposição** | Objeto complexo →  Objeto particionado |  |  |  |
| **Agregação** | Objeto particionado →  Objeto complexo |  |  |  |

1. **(01 ponto)** Escreva um programa em Java que execute as quatro operações aritméticas de números reais
2. **(01 ponto)** Escreva um programa JAVA que calcule a média de quatro notas. O programa deve mostrar o valor da média e indicar se foi aprovado ou reprovado
3. **(01 ponto)** Escreva um programa JAVA que calcule a soma dos inteiros maiores o iguales a um número inteiro A e menores ou iguais a um inteiro B.
4. **(2 pontos)** Para cada caso da questão 2, escrever um programa JAVA, mostrando as classes, as operações (métodos), a criação de objetos e outras características da Programação Orientada a Objetos.
5. **(02 pontos)** Resolver a Questão **9.10** e **9.11** (pág.363) do livro “Introduction to JAVA Programming” , Y. Daniel Liang, 2014. Não precisa desenhar os diagramas UML. “Implemente a classe” = “Faça um programa em Java”

**O professor**

