**TRABALHO DE PROGRAMAÇÃO ORIENTADA A OBEJTOS**

**1**: **Considerando os conceitos básicos de Orientação a Objetos, explique o significado dos conceitos abaixo:**

⠀⠀**Encapsulamento:**

\*Tem o princípio de dividir tarefas em um código de maneira organizada e segura.

\*Evita de ter códigos repetidos. Pôr o código ficar "abstraído".

\*O código ficar "abstraído", por meio de uma interface.

\*Evitando ter contado direto com o algoritmo em questão, assim dando a segurança de n interferir na sua função.

\*Por ter tarefas divididas, a manutenção do código é mais eficiente.

\*O encapsulamento consiste em separar as tarefas em "métodos".

\*Com isso sempre que for realizar uma tarefa, será necessário apenas chamar o método.

⠀⠀**Herança:**

\*A **herança** permite que Classes compartilhem características e ações com outras.

\*Classe que vai compartilhar: Se chama base;

\*Classe que vai receber: Se chama derivada;

**exemplo:**

\*Se uma Classe com nome Pessoa que possui:

\*características: nome e sobrenome;

\*ações: andar e falar;

\*A Classe que recebe herança de Pessoa também vai ter nome e sobrenome,andar e falar.

**Polimorfismo:**

\***Polimorfismo** permite ao desenvolvedor usar o mesmo elemento de formas diferentes

exemplo:

\*Estudar para a prova da faculdade tem várias formas de fazer essa ação:

\*Estudando pra saber resolver a prova.

\*Chutando toda as opções.

\*Fazer de forma errada

\*Temos o mesmo nome pra um comportamento ( Fazer a prova) e várias formas de fazer.

**Atributos é Metodos:**

\***Atributos** são as características de objetos como por exemplo:

\*Avião -> tem roda,volante,banco.

\*Janela -> tem cor,material,espessura.

roda, volante, banco, cor, material eespessura são atributos.

\***Métodos** são ações que o objeto pode fazer:

\*Moto -> acelerar,frear,ligar,desligar.

\*Janela -> abrir,fechar.

acelerar,frear,ligar,desligar,abrir e fechar são métodos

**Atributos e Métodos Estáticos:**

\***Atributo estático** é um estado que vive na classe. **Atributos e Métodos Estáticos** são ações e características que são de uso exclusivo da Classe.

**2:** **Diferencia os mecanismos de sobrecarga e sobrescrita, demonstrando cenário de uso de cada um desses:**

\***Sobrescrita** de métodos é quando estamos trabalhando com orientação a objetos e temos uma classe (filha) que acaba herdando de uma outra classe(mãe). Caso a classe filha precise modificar alguns comportamentos herdado da classe mãe, podemos fazer uma sobrescrita do método diretamente da classe filha

\***A sobrecarga de métodos** é quando temo vários métodos com o mesmo nome, mas com suas assinaturas diferentes, exemplo argumentos diferentes.

**3: A partir do diagrama abaixo, implemente a estrutura de classes.**

Diagrama

Descrição gerada automaticamente

**Pessoa:**

public class Pessoa {

protected String nome;

public Pessoa(String nome) {

this.nome = nome;

}

public void setNome(String nome) {

this.nome = nome;

}

public String getNome() {

return nome;

}

public String toString() {

return "\n" + nome;

}

**Pessoa Física:**

public class PessoaFisica extends Pessoa {

protected String cpf,sobrenome;

protected String tipo = "Pessoa tipo Fisica";

public PessoaFisica(String nome, String sobrenome, String cpf) {

super(nome);

this.cpf = cpf;

this.sobrenome = sobrenome;

}

public String getSobrenome() {

return sobrenome;

}

public void setSobrenome(String sobrenome) {

this.sobrenome = sobrenome;

}

public String getCpf() {

return cpf;

}

public void setCpf(String cpf) {

this.cpf = cpf;

}

public String toString() {

return "\n"

+ tipo

+"\nNome: " + this.getNome()

+"\nSobrenome: " + this.getSobrenome()

+"\nCpf: " + this.getCpf();

}

}

**Pessoa Juridica:**

public class PessoaJuridica extends Pessoa {

protected String cnpj,razaoSocial;

protected String tipo = "Pessoa tipo juridica";

public PessoaJuridica(String nome, String cnpj, String razaoSocial) {

super(nome);

this.cnpj = cnpj;

this.razaoSocial = razaoSocial;

}

public String getRazaoSocial() {

return razaoSocial;

}

public void setRazaoSocial(String razaoSocial) {

this.razaoSocial = razaoSocial;

}

public String getCnpj() {

return cnpj;

}

public void setCnpj(String cnpj) {

this.cnpj = cnpj;

}

public String toString() {

return "\n"

+ tipo

+"\nNome: " + this.getNome()

+"\nSobrenome: " + this.getCnpj()

+"\nCpf: " + this.getRazaoSocial();

}

}

⠀⠀

⠀⠀

⠀⠀

⠀⠀