

Exercícios 1:

- a) A classe é a unidade básica de trabalho em um programa orientado a objetos que representa um modelo abstrato a partir do qual são criadas instâncias (objetos).
- b) O objeto é uma representação real da classe que ocupa espaço na memória e consome recursos do computador.
- c) São dados (simples ou compostos) que caracterizam os objetos daquela classe e são armazenadas em variáveis.
- d) São as operações (procedimentos ou funções) que manipulam os dados.
- e) A assinatura de um método é a forma de identificar um método de forma única. Com por exemplo, a partir do número de parâmetros que um método tem.
- f) A palavra void significa que o método não tem retorno. Alguns métodos não tem essa definição porque elas podem ter algum retorno.
- g) Um construtor de classe é uma função de membro especial de uma classe que é executada sempre que criamos novos objetos dessa classe. Um construtor terá exatamente o mesmo nome que a classe e não possui nenhum tipo de retorno, nem mesmo vazio.
- h) A função main serve como o ponto de partida para a execução do programa.
- i) É um método de mesmo nome com assinaturas diferentes (tipos de argumentos/parâmetros diferentes).
- j) Os métodos static ou métodos de classe são funções que não dependem de nenhuma variável de instância. O modificador static pode ser usado com dados e funções que não precisam de uma instância de classe para se acessada.
- k) A diferença é que o atributo na classe é compartilhado com todas as instâncias. O atributo em uma instância é exclusivo para essa instância.
- l) O escopo de uma variável é o nome que damos a uma utilização de uma variável em um programa.
- m) Eles servem para definir a visibilidade que determinada classe ou membro terá diante das outras.

Exercícios 2:

```
import java.util.Scanner;
```

```
public class Exercicio2 {  
    public static void main(String[] args) {  
  
        int sai, x = 0;  
  
        Cadastro moradores = new Cadastro();
```

```
Condominio[] dados = new Condominio[10];
```

```
Scanner input = new Scanner(System.in);
```

```
for (int i = 0; i < 10; i++) {
```

```
    dados[i] = new Condominio();
```

```
    moradores.cadastro(dados[i]);
```

```
    x++;
```

```
    System.out.println("Quer cadastra mais? 1 - sim ou 2 - não");
```

```
    sai = input.nextInt();
```

```
    if (sai == 2) {
```

```
        break;
```

```
    }
```

```
}
```

```
for (int i = 0; i < x; i++) {
```

```
    System.out.println(dados[i] + "\n");
```

```
}
```

```
}
```

```
}
```

```
class Condominio {
```

```
    String nome, sexo;
```

```
    long cpf, celular;
```

```
    int dataNascimento, bloco, apartamento, codigoAcesso;
```

```
    public String toString() {
```

```
        return "Nome da pessoa: " + nome + "\nCPF:" + cpf + "\nNúmero celular: " + celular  
+ "\nData de nascimento: " + dataNascimento
```

```
        + "\nSexo: " + sexo + "\nBloco: " + bloco + "\nNúmero apartamento: " +  
apartamento + "\nCódigo de acesso: " + codigoAcesso;
```

```
    }
```

```
}
```

```

class Cadastro {

    Scanner input = new Scanner(System.in);

    public void cadastro(Condominio pessoa) {
        System.out.println("Digite o primeiro nome da pessoa:");
        pessoa.nome = input.next();
        System.out.println("Digite o CPF da pessoa:");
        pessoa.cpf = input.nextLong();
        System.out.println("Digite o número de celular da pessoa:");
        pessoa.celular = input.nextLong();
        System.out.println("Digite a data de nascimento da pessoa:");
        pessoa.dataNascimento = input.nextInt();
        System.out.println("Digite o sexo da pessoa:");
        pessoa.sexo = input.next();
        System.out.println("Digite o bloco que a pessoa mora:");
        pessoa.bloco = input.nextInt();
        System.out.println("Digite o número do apartamento:");
        pessoa.apartamento = input.nextInt();
        System.out.println("Digite o código de acesso:");
        pessoa.codigoAcesso = input.nextInt();
    }

}

```

Exercícios 3:

```

public class Exercicio3 {

    public static void main(String[] args) {

        Robo primeiro = new Robo(500);
    }
}

```

```
primeiro.identificacao = 1;
```

```
System.out.println("1 - Ligando o robo:");  
primeiro.ligar();
```

```
System.out.println(primeiro + "\n");
```

```
System.out.println("1 - Movendo o robo:");  
primeiro.andar(5, 10);
```

```
System.out.println(primeiro + "\n");
```

```
System.out.println("1 - Aspirar o pó:");  
primeiro.aspirar(237);
```

```
System.out.println(primeiro + "\n");
```

```
System.out.println("1 - Para o robo:");  
primeiro.parar();
```

```
System.out.println(primeiro + "\n");
```

```
System.out.println("1 - Desligar o robo:");  
primeiro.desligar();
```

```
System.out.println(primeiro + "\n");
```

```
Robo segundo = new Robo(1200);
```

```
segundo.identificacao = 2;
```

```
System.out.println("2 - Ligando o robo:");
```

```
segundo.ligar();
```

```
System.out.println(segundo + "\n");
```

```
System.out.println("2 - Movendo o robo:");
```

```
segundo.andar(32, 20);
```

```
System.out.println(segundo + "\n");
```

```
System.out.println("2 - Aspirar o pó:");
```

```
segundo.aspirar(533);
```

```
System.out.println(segundo + "\n");
```

```
System.out.println("2 - Para o robo:");
```

```
segundo.parar();
```

```
System.out.println(segundo + "\n");
```

```
System.out.println("2 - Desligar o robo:");
```

```
segundo.desligar();
```

```
System.out.println(segundo + "\n");
```

```
}
```

```
}
```

```
class Robo {
```

```
    int identificacao, posicaoX, posicaoY;
```

```
    double limiteDePo, quantidadeDePo;
```

```
    String status;
```

```
    Robo(double limite) {
```

```

        this.status = "Desligado";
        this.posicaoX = 0;
        this.posicaoY = 0;
        this.quantidadeDePo = 0;
        this.limiteDePo = limite;
    }

    public void ligar() {
        this.status = "Ligado";
    }

    public void desligar() {
        this.status = "Desligado";
    }

    public void aspirar(double poAspirar) {
        if (this.status == "Desligado") {
            System.out.println("Ligue o aspirador primeiro");
        } else if (poAspirar + quantidadeDePo > limiteDePo) {
            System.out.println("Utrapassou o limite de Pó");
        } else {
            quantidadeDePo += poAspirar;
        }
    }

    public void andar(int novaX, int novaY) {
        if (this.status == "Desligado") {
            System.out.println("Ligue o aspirador primeiro");
        } else {
            this.status = "Andando";
            this.posicaoX += novaX;
            this.posicaoY += novaY;
        }
    }

```

```

    }
}

public void parar() {
    if (this.status == "Desligado") {
        System.out.println("Ligue o aspirador primeiro");
    } else {
        this.status = "Parado";
    }
}

public String toString() {
    return "Identificação: " + identificacao
        + "\nPosição X: " + posicaoX
        + "\nPosição Y: " + posicaoY
        + "\nQuantidade de pó: " + quantidadeDePo
        + "\nLimite de pó: " + limiteDePo
        + "\nStatus: " + status;
}
}

```

Exercícios 4:

```

import java.util.ArrayList;

public class Exercicio4 {
    public static void main(String[] args) {
        Morador mor1 = new Morador();

        mor1.setNome("Igor");
        mor1.setSexo('M');
        mor1.setCpf("7023182234");
        mor1.setBloco(742);
        mor1.setCelular("6298423423");
    }
}

```

```
mor1.setApartamento(302);
mor1.setCodigoAcesso("8431");
mor1.setDataNascimento(12052001);

Apartamento ap1 = new Apartamento();

ap1.adicionaMorador(mor1);
ap1.setNumero(21);
ap1.setAndar(4);
ap1.setMetragem(500);
ap1.setSituacao("Alugado");

Morador mor2 = new Morador();

mor2.setNome("Regina");
mor2.setSexo('F');
mor2.setCpf("702342832");
mor2.setBloco(239);
mor2.setCelular("52023482321");
mor2.setApartamento(291);
mor2.setCodigoAcesso("391");
mor2.setDataNascimento(21091992);

Apartamento ap2 = new Apartamento();

ap2.adicionaMorador(mor2);
ap2.setNumero(53);
ap2.setAndar(6);
ap2.setMetragem(500);
ap2.setSituacao("Quitado");

String mensagem = ap1 + "\n\n" + ap2;
```



```
System.out.println(mensagem);
```

```
}
```

```
}
```

```
class Apartamento {
```

```
    private int numero, andar;
```

```
    private double metragem;
```

```
    private String situacao;
```

```
    ArrayList<Morador> moradores = new ArrayList<Morador>();
```

```
    public void setNumero(int numero) {
```

```
        this.numero = numero;
```

```
    }
```

```
    public void setAndar(int andar){
```

```
        this.andar = andar;
```

```
    }
```

```
    public void setMetragem(double metragem) {
```

```
        this.metragem = metragem;
```

```
    }
```

```
    public void setSituacao(String situacao) {
```

```
        this.situacao = situacao;
```

```
    }
```

```
    public void adicionaMorador(Morador mor) {
```

```
        moradores.add(mor);
```

```
    }
```

```
    public String toString() {
```

```
String retorno = "Numero: " + numero
                + "\nAndar: " + andar
                + "\nMetragem: " + metragem
                + "\nSituação: " + situacao;
for (int i = 0; i < moradores.size(); i++) {
    retorno = retorno + " " + moradores.get(i);
}
return retorno;
}
}
```

```
class Morador {
    private String nome, cpf, celular, codigoAcesso;
    private int dataNascimento, bloco, apartamento;
    private char sexo;

    public void setNome(String nome) {
        this.nome = nome;
    }

    public void setSexo(char sexo) {
        this.sexo = sexo;
    }

    public void setCpf(String cpf) {
        this.cpf = cpf;
    }

    public void setCelular(String celular) {
        this.celular = celular;
    }
}
```

```

public void setDataNascimento(int dataNascimento) {
    this.dataNascimento = dataNascimento;
}

public void setBloco(int bloco) {
    this.bloco = bloco;
}

public void setApartamento(int apartamento) {
    this.apartamento = apartamento;
}

public void setCodigoAcesso(String codigoAcesso) {
    this.codigoAcesso = codigoAcesso;
}

public String toString() {
    return "\nNome da pessoa: " + nome
        + "\nCPF:" + cpf
        + "\nNúmero celular: " + celular
        + "\nData de nascimento: " + dataNascimento
        + "\nSexo: " + sexo
        + "\nBloco: " + bloco
        + "\nNúmero apartamento: " + apartamento
        + "\nCódigo de acesso: " + codigoAcesso;
}
}

```

Exercícios 5:

```

public class Exercicio5 {

    public static void main(String[] args) {

        // Bancanista
    }
}

```

```

CaixaDeBanco bancanista1 = new CaixaDeBanco("Roberto", 234823);
bancanista1.setHorario("8:30");
bancanista1.setTefone("6292349234");
bancanista1.setSalario(1200);
bancanista1.aumentoSalario(10);

System.out.format("Bancanista:\n%s - Salario: %.2f - Tel: %s - Mat: %d - Horario:
%s - Recibemento anual: %.2f\n",
                bancanista1.getNome(), bancanista1.getSalario(),
bancanista1.getTelefone(),
                bancanista1.getMatricula(), bancanista1.getHorario(),
bancanista1.totalRecebimentoAnual());

// Gerente
Gerente gerente1 = new Gerente("Fábio", 29341);
gerente1.setTefone("6242134123");
gerente1.setSalario(3500);
gerente1.setBonificacao(120);
gerente1.setTipo("Comercial");
gerente1.aumentoSalario(20);

System.out.format("Gerente:\n%s - Salario: %.2f - Tel: %s - Mat: %d - Bonificação:
%.2f - Recibemento anual: %.2f\n",
                gerente1.getNome(), gerente1.getSalario(), gerente1.getTelefone(),
                gerente1.getMatricula(), gerente1.getBonificacao(),
gerente1.totalRecebimentoAnual());

// Clientes
Cliente cliente1 = new Cliente("João", "622342341", 31, "21342134-01");
Cliente cliente2 = new Cliente("Aline", "624121294", 21, "23405324-01");
Cliente cliente3 = new Cliente("Paulo", "624236932", 49, "8394322-01");

cliente2.desativar();

System.out.println(cliente1);

```

```
        System.out.println(cliente2);
        System.out.println(cliente3);
    }
}
```

```
class Pessoas {
    protected String nome;
    protected String telefone;

    public String getNome() {
        return this.nome;
    }

    public String getTelefone() {
        return this.telefone;
    }

    public void setNome(String nome) {
        this.nome = nome;
    }

    public void setTefone(String telefone) {
        this.telefone = telefone;
    }
}
```

```
class Funcionario extends Pessoas {
    protected int matricula;
    protected double salario;

    Funcionario(String nome, int matricula) {
        this.nome = nome;
    }
}
```

```

        this.matricula = matricula;
    }

    public void aumentoSalario(double porcetagem) {
        this.salario = (this.salario * (porcetagem/100)) + this.salario;
    }

    public double totalRecebimentoAnual() {
        return this.salario * 12;
    }

    public int getMatricula() {
        return this.matricula;
    }

    public double getSalario() {
        return this.salario;
    }

    public void setMatricula(int matricula) {
        this.matricula = matricula;
    }

    public void setSalario(double salario) {
        this.salario = salario;
    }
}

class CaixaDeBanco extends Funcionario {
    protected String horario;

    CaixaDeBanco(String nome, int matricula) {

```

```

        super(nome, matricula);
    }

    public String getHorario() {
        return this.horario;
    }

    public void setHorario(String horario) {
        this.horario = horario;
    }
}

class Gerente extends Funcionario {
    protected double bonificacao;
    protected String tipo;

    Gerente(String nome, int matricula) {
        super(nome, matricula);
    }

    public double totalRecebimentoAnual() {
        return super.totalRecebimentoAnual() + (bonificacao * 12);
    }

    public double getBonificacao() {
        return this.bonificacao;
    }

    public String getTipo() {
        return this.tipo;
    }
}

```

```

    public void setBonificacao(double bonificacao) {
        this.bonificacao = bonificacao;
    }

    public void setTipo(String tipo) {
        this.tipo = tipo;
    }
}

class Cliente extends Pessoas {
    protected int idade;
    private String cpf;
    protected char status;

    Cliente(String nome, String telefone, int idade, String cpf) {
        this.nome = nome;
        this.telefone = telefone;
        this.idade = idade;
        this.cpf = cpf;
        this.status = 'A';
    }

    public void desativar() {
        if (this.status == 'A') {
            this.status = 'T';
        } else {
            System.out.println("O cliente já está inativo");
        }
    }

    public String getCpf() {
        return this.cpf;
    }
}

```



```
}
```

```
public int getIdade() {  
    return this.idade;  
}
```

```
public char getStatus() {  
    return this.status;  
}
```

```
public void setIdade(int idade) {  
    this.idade = idade;  
}
```

```
public void setStatus(char status) {  
    this.status = status;  
}
```

```
public void setCpf(String cpf) {  
    this.cpf = cpf;  
}
```

```
public String toString() {  
    String mensagem = String.format("%s - Tel: %s - Idade: %d anos - cpf: %s - %s",  
this.nome, this.telefone, this.idade, this.cpf, this.status);  
    return mensagem;  
}
```

```
}
```