**Curso profissional de Técnico de Gestão e Programação de Sistemas Informáticos**

Disciplina de Programação de Sistemas de Informação

Modulo 9 – Conceitos Avançados De Programação

Conceitos relativos à interface de desenvolvimento de aplicações (API) do sistema operativo

Alunos: José Pereira

Professor/a:

Caldas da Rainha, Setembro de 2019

AGRADECIMENTOS

ÍNDICE GERAL

[AGRADECIMENTOS 2](#_Toc20846493)

[ÍNDICE GERAL 3](#_Toc20846494)

[ÍNDICE DE FIGURAS 4](#_Toc20846495)

[ÍNDICE DE QUADROS 5](#_Toc20846496)

[INTRODUÇÃO 6](#_Toc20846497)

[1. Modulo 9 Introdução á programação orientada a objetos(POO) 7](#_Toc20846498)

[1.1. Exercicios Realizado na 1 aula 7](#_Toc20846499)

[1.1.1. Classe Principal 7](#_Toc20846500)

[1.1.2. My class 1 7](#_Toc20846501)

[1.1.3. My class 2 8](#_Toc20846502)

[CONCLUSÃO 9](#_Toc20846503)

[REFERÊNCIAS BIBLIOGRÁFICAS 10](#_Toc20846504)

[APÊNDICES 11](#_Toc20846505)

[ANEXOS 12](#_Toc20846506)

ÍNDICE DE FIGURAS

ÍNDICE DE QUADROS

INTRODUÇÃO

1. Modulo 9 Introdução á programação orientada a objetos(POO)
   1. Exercicios Realizado na 1ª aula
      1. Classe Principal –Primeiro Projeto

import java.util.Scanner;

//ver metodos da classe java.util.Scanner

public class Principal{

public static void main(String[] args) {

int anoAtual,anoNascimento,qtdeAnos;

System.out.println("Entre o ano atual");

anoAtual = new Scanner(System.in).nextInt();

System.out.println("Entre o ano atual");

anoNascimento = new Scanner(System.in).nextInt();

qtdeAnos = anoAtual - anoNascimento;

System.out.println("O Utilizador tem"+qtdeAnos+" anos");

}

}

* + 1. My class 1-Primeiro Projeto

import java.util.Scanner;

class Myclass1 {

Scanner myObj = new Scanner(System.in);

System.out.println("Enter username");

String userName = myObj.nextline();

System.out.println("Username is: "+myObj);

}

* + 1. My class 2 –Primeiro Projeto

import java.util.Scanner;

class Myclass1 {

public static void main (String[] args){

Scanner myObj = new Scanner(System.in);

System.out.println("Enter name, age and salary");

String Name = myObj.nextLine();

int age = myObj.nextInt();

double salary = myObj.nextDouble();

System.out.println("Name: "+Name+"\nAge: "+age+"\nSalary: "+salary);

}

}

* 1. Exercicios Realizados na 2ª aula (Continuação Do Primeiro Projeto)

1.2.1. My class 3 Projeto 1

import java.time.LocalDate;

public class Myclass3{

public static void main(String[] args){

LocalDate myObj= LocalDate.now();

System.out.println(myObj);

}

}

1.2.2. My class 4 Projeto 1

import java.util.Scanner;

import java.time.LocalTime;

public class Myclass4{

public static void main(String[] args){

LocalTime myobj= LocalTime.now();

System.out.println(myobj);

}

}

1.2.3. My class 5 Projeto 1

import java.time.LocalDateTime;

public class Myclass5{

public static void main(String[] args){

LocalDateTime myObj=LocalDateTime.now();

System.out.println(myObj);

}

}

1.2.4. My class 6 Projeto 1

import java.time.LocalDateTime;

import java.time.format.DateTimeFormatter;

public class Myclass6{

public static void main (String[] args){

LocalDateTime myObj= LocalDateTime.now();

System.out.println("Before Formated: "+myObj);

DateTimeFormatter myFormatobj = DateTimeFormatter.ofPattern("dd/mm/yyyy HH:mm:ss");

String FormattedDate = myObj.format(myFormatobj);

System.out.println("After Formatted:"+FormattedDate);

}

}

1.2.5. My Class Projeto 1

public class Myclass{

int x = 5;

public static void main(String[] args){

Myclass myObj = new Myclass();

//myObj.x = 25;

System.out.println(myObj.x);

Myclass myObj1 = new Myclass();

Myclass myObj2 = new Myclass();

System.out.println(myObj1.x);

System.out.println(myObj2.x);

}

}

**1.2.6. Other Object Projeto 1**

public class Otherclass{

public static void main (String[] args){

Myclass myObj= new Myclass();

System.out.println(myObj.x);

}

}

1.2.7. Começo de um novo projeto –Projeto 0/My class 1

public class Myclass{

/\*\*Static Define o método como estático isso quer dizer que a classe nao precisa ser

\* instanciada para chamar este metodo

\* void é tipo de retorno do metodo este tipo de retorno significa vazio.

\*/

public static void main(String[] args){

System.out.println("Hello World");

}

}

1.2.8 Variáveis Projeto 0

public class Variaveis{

/\*String Armazena texto como "Hello". Os valores das cadeis sao separadas por aspas duplas

\* int aramazena numeros inteiros

\* char armazena caracteres

\* float armazena numeros de ponto flutuante com decimais como 19.12

\* booleano armazena valores de TRUE or FALSE ou entao 0 ou 1

\*

\*/

int x = 5, y=15;

char z = 'a';

String firstname="Jose";

String lastname="Pereira";

String fullname=firstname + lastname;

public Variaveis{

System.out.println(Fullname);

}

public static void main(String[] args){

System.out.println(x+y);

}

public int sampleMethod(int x,int y){

return x + y ;

}

}

1.2.9. MyClassNarrowing

public class MyclassRidening{

public static void main(Strings[] args){

double myDouble = 9.78;

int myint =(int) myDouble;

System.out.println(myDouble);

system.out.println(myint);

}

}

* 1. Exercícios Realizados na 3ª aula (Exemplo 2)

1.3.1. Principal

import java.util.Scanner;

public class Principal

{

public static void main(String[] args){

double nota1,nota2,nota3,media;

System.out.println("Entre com a primeira nota: ");

nota1 = new Scanner(System.in).nextDouble();

System.out.println("Entre com a segunda nota: ");

nota2 = new Scanner(System.in).nextDouble();

System.out.println("Entre com a terceira nota: ");

nota3 = new Scanner(System.in).nextDouble();

System.out.println("Media de : "+(nota1+nota2+nota3)/3);

}

}

* 1. Exercicios Realizados na 4ª aula

1.4.1. Exemplo 4 IdadeAdulta

import java.util.Scanner;

public class IdadeAdulta{

public static void main(String[] args){

Scanner ler = new Scanner(System.in);

int n=10;

int idade[]=new int[n];

int i;

int menor,posmenor,maior,posmaior;

for(i=0;i<n;i++)

{

System.out.printf("Informe a idade da %2da, pessoa de %d: ",(i+1),n);

idade[i]= ler.nextInt();

}

menor =idade[0];posmenor = 0;

maior=idade[0];posmaior=0;

for(i=0;i<n;i++){

if(idade[i]<menor){

menor=idade[i];

posmenor = i;

}

else if(idade[i]>maior){

maior=idade[i];

posmaior=i;

}

}

System.out.printf("\nA maior Idade é de %d e encontra-se na posição %d",maior,(posmaior+1));

System.out.printf("\nA menor Idade é de %d e encontra-se na posição %d\n",menor,(posmenor+1));

}

}

1.4.2 Exercicio 1(classe Pessoa)

/\*\*

\* Escreva a descrição da classe Pessoa aqui.

\*

\* @author (seu nome)

\* @version (número de versão ou data)

\*/

public class Pessoa

{

//Atributos da classe PEssoa(Estado)

String nome;

char genero;

int idade;

/\*\*

\* COnstrutor para classe Pessoa default

\*/

public Pessoa(){

}

public Pessoa(String n,char g)

{this.nome=n;

this.genero=g;

}

public Pessoa(String n,char g,int i){

this(n,g);

this.idade=i;

}

//Métodos (Comportamentos)

void falar(){

System.out.println("Blá,Blá...");

}

}

1.4.3. Exercicio 1(Classe UsaPessoa)

import java.util.Scanner;

public class UsaPessoa{

public UsaPessoa(){}

public static void main(String[] args){

Scanner s = new Scanner(System.in);

Pessoa pessoa1;

pessoa1 = new Pessoa();

pessoa1.nome= "Camilla";

pessoa1.genero='F';

pessoa1.idade= 12;

System.out.println("O objeto pessoal chama-se "+pessoa1.nome+"E do sexo: "+pessoa1.genero+" com "+pessoa1.idade+" anos de idade e diz o seguinte");

pessoa1.falar ();

Pessoa pessoa2;

pessoa2 = new Pessoa();

pessoa2.nome= "Esdrubal";

pessoa2.genero='M';

pessoa2.idade= 67;

System.out.println("\nO objeto pessoal chama-se "+pessoa2.nome+"E do sexo: "+pessoa2.genero+" com "+pessoa2.idade+" anos de idade e diz o seguinte");

pessoa2.falar ();

Pessoa pessoa3;

System.out.println("\nIndique o nome da Pessoa: ");

String n = s.nextLine();

System.out.println("\nIndique o Sexo (F/M): ");

String gen = s.nextLine();

char g=gen.charAt(0);

System.out.println("\nIndique a idade : ");

int i =s.nextInt();

pessoa3 = new Pessoa(n,g,i);

System.out.println("O objeto Pessoa3 chama-se "+pessoa3.nome+"É do Sexo "+pessoa3.genero+" e tem " +pessoa3.idade+"anos e diz o seguinte: ");

pessoa3.falar();

}

}

**1.4.4. Exercicio 1 (Pessoa)**

/\*\*

\* Escreva a descrição da classe Pessoa aqui.

\*

\* @author (seu nome)

\* @version (número de versão ou data)

\*/

public class Pessoa

{

//Atributos da classe PEssoa(Estado)

String nome;

char genero;

int idade;

/\*\*

\* COnstrutor para classe Pessoa default

\*/

public Pessoa(){

}

public Pessoa(String n,char g)

{this.nome=n;

this.genero=g;

}

public Pessoa(String n,char g,int i){

this(n,g);

this.idade=i;

}

//Métodos (Comportamentos)

void falar(){

System.out.println("Blá,Blá...");

}

}

* 1. Exercicios Realizados na aula 6

1.5.1 Classe Carro

public class Carro

{

//Atributos

private String matriculas;

private int velocidadeatual =0;

private int velocidademaxima =200;

private Condutor condutor;

/\*\*

\* COnstrutor para objetos da classe Carro

\*/

public Carro()

{

}

/\*\*

\* Exemplo de método - substitua este comentário pelo seu próprio

\*

\* @param y exemplo de um parâmetro de método

\* @return a soma de x com y

\*/

void ligar(){

System.out.println("VRUUUUUUUUUMMMMMMMM");

}

void desligar(){

System.out.println("MMMMMMMMMMMMM");

}

void acelarar(){

/\*pergunta 3\*/velocidadeatual+=10+condutor.destreza\*0.1;

if(velocidadeatual >velocidademaxima)

velocidadeatual=velocidademaxima;

}

void travar (int intensidadetravagem){

if(intensidadetravagem>velocidademaxima)

intensidadetravagem = velocidademaxima;

else

if(intensidadetravagem<0)

intensidadetravagem=0;

velocidadeatual =intensidadetravagem;

if(velocidadeatual>velocidademaxima)

velocidadeatual=velocidademaxima;

else if(velocidadeatual<0)

velocidadeatual=0;

}

public String getMatricula(){

return matriculas;

}

public void setMatricula(){

this.matriculas= matriculas;

}

public Condutor getCondutor(){

return condutor;

}

public void seCondutor(Condutor condutor){

this.condutor= condutor;

}

public int getVelocidadeatual(){

return velocidadeatual;

}

public int getVelocidadeMaxima(){

return velocidademaxima;

}

public void setVelocidadeMaxima(){

this.velocidademaxima=velocidademaxima;

}

}

1.5.2 Classe Condutor

public class Condutor{

public String nome;

public int idade;

protected int destreza;

public String getNome() {

return nome;

}

public void setNome(String nome){

this.nome=nome;

}

public int getIdade (){

return idade;}

public void setIdade (int idade){

this.idade=idade;

}

public int getDestreza(){

return destreza;

}

public void setDestreza(int destreza){

this.destreza=destreza;

}

public Condutor(){

}

}

1.5.3 Classe Corrida

import java.util.Scanner;

public class Corrida{

//Construtores

public Corrida(){

}

//7Métodos

public static void main(String[] args){

/\*

\*Cria objetos carro1 e carro2 que são instancias da classe carro

\*/

Carro carro1= new Carro();

Carro carro2= new Carro();

/\*

\*Define valores para o atributo matrícula dos objetos carro1 e carro2

\*/

carro1.matriculas="11-AA-11";

carro2.matriculas="22-BB-22";

/\*Cria os objetos condutor1 e condutor2 que sao instâncias da classe Condutor

\* e irao conduzir os carros

\*/

Condutor condutor1= new Condutor ();

Condutor condutor2= new Condutor();

/\*Define os valores para os atributos condutor1

\*/

condutor1.nome="Antonio";

condutor1.idade=25;

condutor1.destreza=75;

/\*Define os valores para os atributos condutor2

\*/

condutor1.nome="Manuel";

condutor1.idade=27;

condutor1.destreza=65;

//Coloca os condutores nos seus carros

carro1.condutor=condutor1;

carro2.condutor=condutor2;

//liga carros

carro1.ligar();

carro2.ligar();

//Inicia a corrida

carro1.acelarar ();

carro2.acelarar ();

carro1.acelarar ();

carro2.travar (10);

carro1.acelarar ();

carro1.travar (5);

carro2.acelarar ();

carro2.velocidadeatual=300;

System.out.println("Velocidade do carro 1: "+carro1.velocidadeatual);

System.out.println("Velocidade do carro 2: "+carro2.velocidadeatual);

carro1.desligar();

carro2.desligar();

}

}

1.5.5.Trabalho Projeto Classe Carro Corrida

public class CarroCorrida{

protected Piloto piloto;

protected int velocidadeatual=0;

protected int velocidademaxima=360;

private int i=0;

protected String numeroIdentificacao;

protected Motor motor;

public CarroCorrida(){

}

public String getnumeroIdentificacao(){

return numeroIdentificacao;

}

public void setIdentificacao(){

this.numeroIdentificacao= numeroIdentificacao;

}

public Piloto getPiloto(){

return piloto;

}

public void seCondutor(Piloto piloto){

this.piloto= piloto;

}

public int getVelocidadeatual(){

return velocidadeatual;

}

public int getVelocidadeMaxima(){

return velocidademaxima;

}

public void setVelocidadeMaxima(){

this.velocidademaxima=velocidademaxima;

}

void ligar(){

System.out.println("VRUUUUUUUUUMMMMMMMM");

i=i+1;

}

void desligar(){

System.out.println("MMMMMMMMMMMMM");

i=0;

}

void acelerar(){

if(i==1){

velocidadeatual+=10+piloto.destreza\*0.1+motor.potencia\*0.1;

if(velocidadeatual>velocidademaxima)

velocidadeatual=velocidademaxima;

}

else

System.out.println("O Carro Nao foi Ligado");

}

void travar (int intensidadetravagem){

if(i==1){

if(intensidadetravagem>velocidademaxima)

intensidadetravagem = velocidademaxima;

else

if(intensidadetravagem<0)

intensidadetravagem=0;

velocidadeatual =intensidadetravagem;

if(velocidadeatual>velocidademaxima)

velocidadeatual=velocidademaxima;

else if(velocidadeatual<0)

velocidadeatual=0;

}

else

System.out.println("O Carro Nao foi Ligado");

}

}

1.5.6. Trabalho Projeto Classe Corrida

public class Corrida{

public Corrida(){

}

public static void main(String[] args)

{

CarroCorrida carroEquipa1 =new CarroCorrida();

CarroCorrida carroEquipa2 =new CarroCorrida();

carroEquipa1.numeroIdentificacao="95";

carroEquipa2.numeroIdentificacao="88";

Piloto piloto1= new Piloto();

Piloto piloto2=new Piloto();

Motor motor1= new Motor();

Motor motor2= new Motor();

motor1.potencia= 50;

motor2.potencia=50;

piloto1.nome="Faisca";

piloto1.idade=5;

piloto1.destreza=50;

piloto2.nome="JJBOCE";

piloto2.idade=88;

piloto2.destreza=20;

carroEquipa1.piloto=piloto1;

carroEquipa2.piloto=piloto2;

carroEquipa1.motor=motor1;

carroEquipa2.motor=motor2;

carroEquipa1.ligar();

carroEquipa2.ligar();

carroEquipa1.acelerar ();

carroEquipa2.acelerar ();

carroEquipa1.acelerar ();

carroEquipa2.travar (10);

carroEquipa1.acelerar ();

carroEquipa1.travar (5);

carroEquipa2.acelerar ();

carroEquipa1.acelerar ();

//carroEquipa2.velocidadeatual = 300.0f;

System.out.println("Velocidade do carro 1: "+carroEquipa1.velocidadeatual);

System.out.println("Velocidade do carro 2: "+carroEquipa2.velocidadeatual);

carroEquipa1.desligar();

carroEquipa2.desligar();

}

}

1.5.7 Trabalho Projeto Classe Piloto

public class Piloto{

protected String nome;

protected int idade;

protected int destreza;

public String getNome(){

return nome;

}

public void setNome(String nome){

this.nome=nome;

}

public int getIdade (){

return idade;

}

public void setIdade (int idade){

this.idade=idade;

}

public int getDestreza(){

return destreza;

}

public void setDestreza(int destreza){

this.destreza=destreza;

}

public Piloto(){

}

}

1.5.8 Trabalho projeto Classe Motor

public class Motor{

protected int potencia;

public Motor(){

}

void Potencia(int potencia){

if(potencia>100)

potencia=100;

else if(potencia<1)

potencia=1;

}

}

* 1. Exercicios Realizados na 6 aula

1.6.1 Exercico 3 Classe Contador

public class Contador{

private int cont=0;

private int numero;

public int getnumero(){

return numero;

}

public void setnumero(int num){

this.numero=num;

}

public int getcont(){

return cont;

}

public void setcont(int num){

this.cont=num;

}

public Contador(){

}

public void Conta(int cont){

this.cont=cont;

}

public void Contmais(){

cont=cont+1;

}

public void Contmenos(){

cont=cont-1;

}

public void ContMostrar(){

System.out.println(cont);

}

}

1.6.2.Classe UsaContador1

public class UsaContador1{

public static void main(String[] args)

{

Contador conta= new Contador();

conta.setcont(0);

conta.Contmais();

conta.ContMostrar();

conta.Contmais();

conta.ContMostrar();

conta.Contmenos();

conta.ContMostrar();

}

}

1.6.3 Classe UsaContador2

import java.util.Scanner;

public class UsaContador2{

public UsaContador2(){

}

public static void main(String[] args){

Contador contad= new Contador();

contad.setcont(0);

Scanner s= new Scanner(System.in);

while(contad.getcont()<=5){

System.out.println("Carregue na tecla <<Enter>> para retirar uma senha");

s.nextLine();

contad.Contmais();

contad.ContMostrar();

if(contad.getcont()==3)

{

System.out.println("Carregue com mais papel");

}

}

}

}

1.6.4 Projeto 4 Calculadora

public class Calculadora{

public Calculadora(){

}

public double Dividir(double x,double y){

return x/y;

}

public double Subtrair(double x,double y){

return x-y;

}

public double Somar(double x,double y){

return x+y;

}

public double Multiplicar(double x,double y){

return x\*y;

}

}

1.6.5 Classe FazCalculos1

public class FazCalculos1{

public static void main(String[] args){

Calculadora calc = new Calculadora();

System.out.println("Soma = "+calc.Somar(5.2,2.7));

System.out.println("Subtrair = "+calc.Subtrair(5.2,2.7));

System.out.println("Dividir = "+calc.Dividir(5.2,2.7));

System.out.println("Multiplicação = "+calc.Multiplicar(5.2,2.7));

}

}

1.6.6 Classe FazCalculos2

import java.util.Scanner;

public class FazCalculos2{

private double x;

private double y;

public static void main(String[] args){

Calculadora calc= new Calculadora();

Scanner s = new Scanner(System.in);

System.out.println("Introduza o primeiro numero: ");

double x= s.nextDouble();

System.out.println("Introduza o segundo numero: ");

double y= s.nextDouble();

System.out.println("Soma = "+calc.Somar(x,y));

System.out.println("Subtrair = "+calc.Subtrair(x,y));

System.out.println("Dividir = "+calc.Dividir(x,y));

System.out.println("Multiplicação = "+calc.Multiplicar(x,y));

}

}

* 1. Exercicios Realizados na 7 aula

1.7.1 Exercicio 5 Classe Retangulo

public class Retangulo{

private double x1,x2,y1,y2;

public double getx1(){

return x1;

}

public void setx1(double x1){

this.x1=x1;

}

public double getx2(){

return x2;

}

public void setx2(double x2){

this.x2=x2;

}

public double gety1(){

return y1;

}

public void sety1(double y1){

this.y1=y1;

}

public double gety2(){

return y2;

}

public void sety2(double y2){

this.y2=y2;

}

public Retangulo(){

}

public double Calcbase(){

return x2-x1;

}

public double Calcalt(){

return y2-y1;

}

public double CalcPeri(){

return ((y2-y1)\*2+(x2-x1)\*2);

}

public double CalcArea(){

return (y2-y1)\*(x2-x1);

}

}

1.7.2 Classe ConstroiRetangulo

public class ConstroiRetangulo{

public static void main(String [] args){

Retangulo rect = new Retangulo();

rect.setx1(2.0);

rect.sety1(1.0);

rect.setx2(4.0);

rect.sety2(5.0);

System.out.println("Base: "+rect.Calcbase()+"cm");

System.out.println("Altura: "+rect.Calcalt()+"cm");

System.out.println("Perímetro: "+rect.CalcPeri()+"cm");

System.out.println("Area: "+rect.CalcArea()+"cm2");

}

}

CONCLUSÃO

REFERÊNCIAS BIBLIOGRÁFICAS

APÊNDICES

ANEXOS