



CENTRO UNIVERSITARIO UAEM

ATLACOMULCO

CARRERA

**LICENCIATURA EN INGENIERÍA EN
SISTEMAS COMPUTACIONALES**

MATERIA

PARADIGMAS DE LA PROGRAMACIÓN

DOCENTE

JULIO ALBERTO DE LA TEJA LÓPEZ

ALUMNO

ALEXIS VALENCIA MARTÍNEZ

FECHA DE ENTREGA

4-SEPTIEMBRE-2023

➤ FIGURE.JAVA

```
public abstract class Figure{
    //Name of figure
    protected String Name;
    public Figure(String Name){
        this.Name= Name;
    }
    //methods
    public abstract double calculateArea();
    public abstract double calculatePerim();
}
```

➤ RECTANGLE.JAVA

```
public class Rectangle extends Figure{
    private double base;
    private double length;

    public Rectangle(double base, double length){
        super("Rectangle");
        this.length= length;
        this.base = base;
    }
    @Override
    //Area calculate
    public double calculateArea() {
        return base * length;
    }
    //Perimeter calculate
    public double calculatePerim(){
        return(base+length);
    }
}
```

➤ TRIANGLE.JAVA

```
public class Triangle extends Figure{
    //Add base, height and sides
    private double base;
    private double height;
    private double sideA;
    private double sideB;
    private double hypotnuse;
    public Triangle(double base, double height,double sideA,double sideB,double
hypotnuse)throws Exception{
        super("Triangle");
        //Failure to comply will result in an error
        if (sideA + sideB <= hypotnuse || sideA + sideB <= hypotnuse || sideB + hypotnuse <=
sideA) {
            throw new Exception("Error: re-enter the sides ");
        }
    }
}
```

```

    }
    this.base = base;
    this.height= height;
    this.sideA= sideA;
    this.sideB= sideB;
    this.hypotnuse= hypotnuse;
}
@Override
//Area calculate
public double calculateArea() {
    double area = 0;
    try {
        area = (base * height) / 2;
    } catch (ArithmeticException e) {
        System.out.println("Error: División entre cero");
    }
    return area;
}
//perimeter calculate
public double calculatePerim(){
    return(sideA+sideB+hypotnuse);
}
}

```

➤ CIRCLE.JAVA

```

public class Circle extends Figure{
    private double radius;
    public Circle(double radius){
        super("Circle");
        this.radius=radius;
    }
    @Override
    //Area calculate
    public double calculateArea() {
        return Math.PI * Math.pow(radius, 2);
    }
    //perimeter calculate
    public double calculatePerim(){
        return Math.PI*(radius+radius);
    }
}

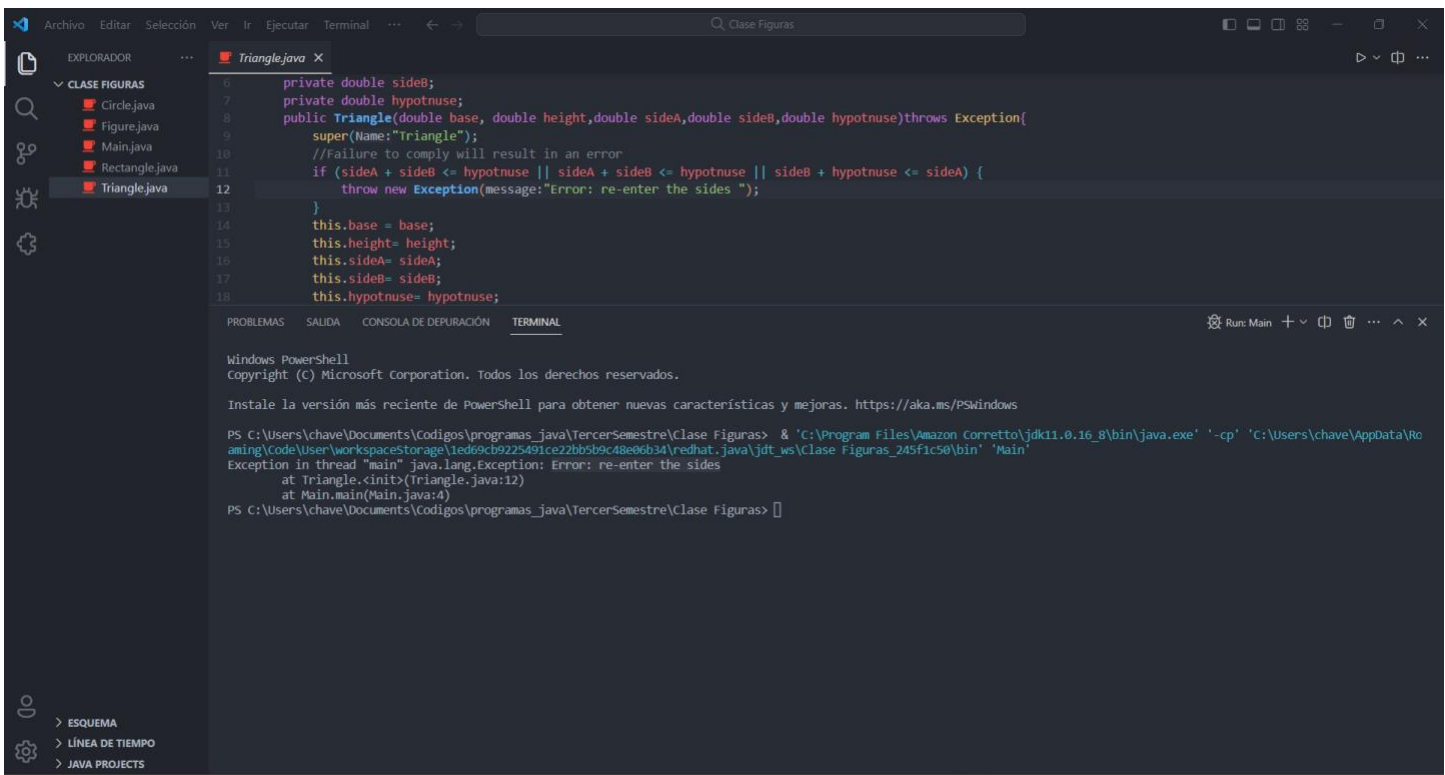
```

➤ MAIN.JAVA

```
public class Main {  
    public static void main(String[] args) throws Exception {  
        Rectangle rectangle = new Rectangle(4.0, 5.2);  
        Triangle triangle = new Triangle(10, 10,0,0,10);  
        Circle circle = new Circle(10);  
  
        System.out.println("Area of "+rectangle.Name + ": "+  
rectangle.calculateArea()+" Perimeter: "+ rectangle.calculatePerim());  
        System.out.println("Area of "+triangle.Name + ": "+  
triangle.calculateArea()+" Perimeter: "+ triangle.calculatePerim());  
        System.out.println("Area of "+circle.Name + ": "+  
circle.calculateArea()+" Perimeter: "+ circle.calculatePerim());  
    }  
}
```

- In which figure can this error occur?
triangle would be the closest
- How did you solve it?
Adding an exception that finds out when it divides by 0

➤ SALIDA



The screenshot shows an IDE with the following components:

- EXPLORADOR:** A sidebar on the left showing a project structure with files: Circle.java, Figure.java, Main.java, Rectangle.java, and Triangle.java. Triangle.java is selected.
- Triangle.java:** The main editor shows the code for the Triangle class. It includes private fields for sideB, hypotnuse, and public fields for base, height, sideA, and sideB. The constructor takes five arguments and throws an Exception. A comment indicates that failure to comply will result in an error. A conditional statement checks for division by zero and throws an Exception with the message "Error: re-enter the sides " if the condition is met.
- TERMINAL:** The bottom panel shows the output of the program. It displays the Windows PowerShell prompt, the command to run the program, and the resulting output, which includes the area and perimeter of the triangle and a runtime exception: "java.lang.Exception: Error: re-enter the sides " at Main.main(Main.java:4).

➤ CHARACTER.JAVA

```
public class Character {
    protected String Name;
    protected int Level;
    public Character(String Name, int level){
        if (level < 1) {
            // Error in case of level is minor than 1
            throw new IllegalArgumentException("Level is minor than 1");
        }
        this.Name = Name;
        this.Level = level;
    }
    public void Attack(int level){
        if (level < 5) {
            // Error in case of level is minor than 1
            throw new RuntimeException("Level is minor than 5, you cannot attack");
        }else{
            System.out.println("You are attacking!");
        }
    }
    public void displayAttributes() {
        System.out.println("Name: " + Name);
        System.out.println("Level: " + Level);
    }
}
```

➤ ENEMY.JAVA

```
public class Enemy extends Character{
    protected String type;

    public Enemy(String Name, int Level,String Enemy){
        super("Boss Final", 99);
        this.type = Enemy;
    }
    public void Scream(){
        System.out.println(Name + "Scream!!!");
    }
    @Override
    public void displayAttributes(){
        super.displayAttributes();
        System.out.println("Type: "+ type);
    }
}
```

✓ PLAYER.JAVA

```
public class Player extends Character{
    protected String Class;
    public Player(String Name,int level,String Class){
        super("Jose", 12);
        this.Class= Class;
    }
    public void useAbility(Enemy enemy){
        System.out.println("Use special ability vs "+ enemy.Name);
    }

    @Override
    public void displayAttributes(){
        super.displayAttributes();
        System.out.println("Class: "+ Class);
    }
}
```

➤ MAIN.JAVA

```
import java.util.Random;

public class Main {
    public static void main(String[] args) {
        Player player = new Player("Jorge", 50, "Mago");
        Enemy Zombie = new Enemy("Zombie", 23, "Undead");
        Enemy Boss = new Enemy("Medusa", 90, "Boss");
        Enemy Boss_Final = new Enemy("Dragon", 90, "Boss Final");
        player.displayAttributes();
        player.Attack(50);

        Random random = new Random();
        int RandomX = random.nextInt(3);
        Enemy randomEnemy;

        if(RandomX==0){
            randomEnemy = Boss_Final;
        }else if(RandomX == 1){
            randomEnemy = Boss;
        }else{
            randomEnemy = Zombie;
        }
        player.useAbility(randomEnemy);

        System.out.println();
        Zombie.displayAttributes();
        Zombie.Attack(23);
        Zombie.Scream();

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

```

    Boss.displayAttributes();
    Boss.Attack(90);
    Boss.Scream();

    System.out.println();
    Boss_Final.displayAttributes();
    Boss_Final.Attack(2);
    Boss_Final.Scream();
}
}

```

➤ SALIDA

The screenshot shows an IDE with a project named 'JERARQUIA'. The file explorer on the left lists 'Character.java', 'Enemy.java', 'Main.java', and 'Player.java'. The editor displays the 'Main.java' file with the following code:

```

1  import java.util.Random;
2
3  public class Main {
4      public static void main(String[] args) {
5          Player player = new Player(Name:"Jorge", Level:50, Class:"Mago");
6          Enemy Zombie = new Enemy(Name:"Zombie", Level:23, Enemy:"Undead");
7          Enemy Boss = new Enemy(Name:"Medusa", Level:90, Enemy:"Boss");
8          Enemy Boss_Final = new Enemy(Name:"Dragon", Level:2, Enemy:"Boss Final");
9          player.displayAttributes();
10         player.Attack(level:50);
11
12         Random random = new Random();

```

The terminal window at the bottom shows the output of the program:

```

PS C:\Users\chave\Documents\Codigos\programas_java\TercerSemestre\Jerarquia> & 'C:\Program Files\Amazon Corretto\jdk11.0.16_8\bin\java.exe' '-cp' 'C:\Users\chave\AppData\Roamin
g\Code\User\workspaceStorage\3a2fb99286d449dde8185bea59ed50be\redhat.java\jdt_ws\Jerarquia_260f457\bin' 'Main'
Name: Jose
Level: 12
Class: Mago
You are attacking!
Use special ability vs Boss Final

Name: Boss Final
Level: 99
Type: Undead
You are attacking!
Boss FinalScream!!!

Name: Boss Final
Level: 99
Type: Boss
You are attacking!
Boss FinalScream!!!

Name: Boss Final
Level: 99
Type: Boss Final
Exception in thread "main" java.lang.RuntimeException: Level is minor than 5, you cannot attack
    at Character.Attack(Character.java:15)
    at Main.main(Main.java:37)
PS C:\Users\chave\Documents\Codigos\programas_java\TercerSemestre\Jerarquia>

```

➤ PALETTE.JAVA

```
public class palette{
    protected String Flavor;
    protected double price;

    public palette(String Flavor, double price){
        //In case it is marked that it is one of the two types, it will give an error
        if (!Flavor.equals("Water") && !Flavor.equals("Cream")) {
            throw new IllegalArgumentException("The flavor is Water or Cream");
        }
        this.Flavor= Flavor;
        this.price= price;
    }

    public void showInfo(){
        System.out.println("Flavor is: "+ Flavor);
        System.out.println("Price is: "+ price);
    }
    public void discount(double percentage){
        double discount= price* (percentage/100);
        price = price - discount;
        System.out.println("Discount Applied is: " + discount);
    }
}
```

➤ PCREAM.JAVA

```
public class PCream extends palette{
    protected Boolean cream;
    public PCream(String Flavor,Double price,boolean cream){
        super("Crem", 30);
        this.cream= cream;
    }
    public void showCream(){
        System.out.println("Cream texture: "+(cream ? "Yes": "No"));
    }
    public void ChangePrice(){
        price= price +6;
    }
}
```

➤ PWATER.JAVA

```
public class PWater extends palette{
    protected boolean BaseWater;
    public PWater(String flavor, double Price,boolean Base){
        super("Agua", 20.0);
        this.BaseWater = Base;
    }
}
```



```

public void showBaseW(){
    System.out.println("Water Base: "+(BaseWater ? "Yes": "No"));
}
public void ChangePrice(){
    price = price+2;
}
}

```

➤ MAIN.JAVA

```

import java.util.Random;

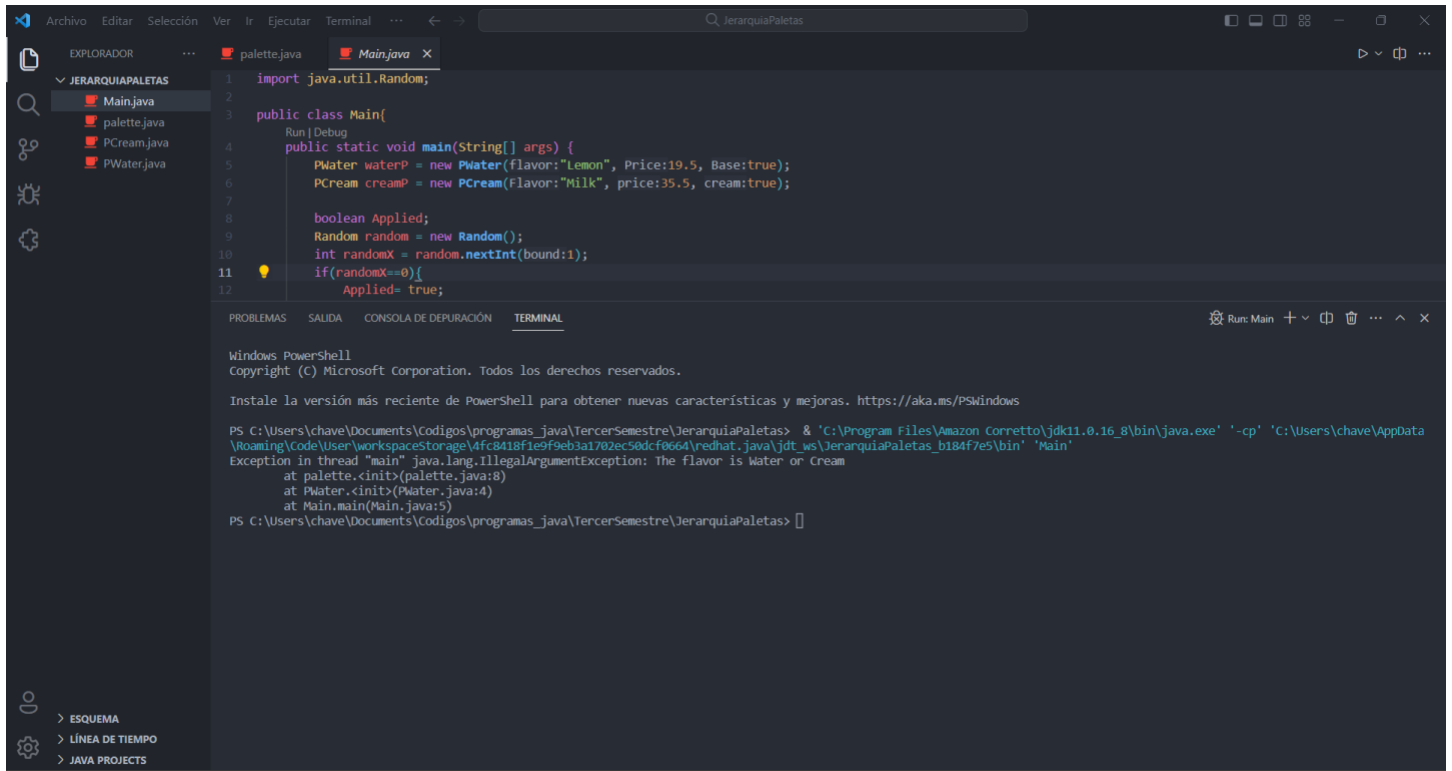
public class Main{
    public static void main(String[] args) {
        PWater waterP = new PWater("Lemon", 19.5, true);
        PCream creamP = new PCream("Milk", 35.5, true);

        boolean Applied;
        Random random = new Random();
        int randomX = random.nextInt(1);
        if(randomX==0){
            Applied= true;
        }else{
            Applied=false;
        }

        waterP.showInfo();
        waterP.showBaseW();
        waterP.ChangePrice();

        creamP.showInfo();
        creamP.showCream();
        creamP.ChangePrice();
        if(Applied == true){
            System.out.println("Congratulations!!!!\n"+"You win discount in your purchase");
            waterP.discount(20);
            creamP.discount(10);
        }
    }
}

```



PlantUML Web Server

https://www.plantuml.com/plantuml/uml/SoWklmgAStDuNBEIImkDZ1KiAdHrLM0S8oWWiOAMd0n4wYOGKB-NcmCacQkeAS75RA02ba...

```

+calculateArea()
+calculatePerim()
}

class Circle {
  -radius: double
  +Circle(radius)
  +calculateArea()
  +calculatePerim()
}

class Main {
  +main(args)
}

Rectangle --|> Figure
Triangle --|> Figure
Circle --|> Figure
@enduml

```

report this ad

UML Class Diagram:

- Rectangle** (Class):
 - Attributes: base: double, length: double
 - Operations: Rectangle(base, length), calculateArea(), calculatePerim()
- Triangle** (Class):
 - Attributes: base: double, height: double, sideA: double, sideB: double, hypotnuse: double
 - Operations: Triangle(base, height, sideA, sideB, hypotnuse), calculateArea(), calculatePerim()
- Circle** (Class):
 - Attributes: radius: double
 - Operations: Circle(radius), calculateArea(), calculatePerim()
- Main** (Class):
 - Operations: main(args)
- Figure** (Class):
 - Attributes: Name: String
 - Operations: Figure(Name), calculateArea(), calculatePerim()

Relationships: Rectangle, Triangle, and Circle are subclasses of Figure.

PlantUML Web Server

https://www.plantuml.com/plantuml/uml/SoWklmgAStDuNBEIImkDZ1KiAdHrLM0S8oWWiOAMd0n4wYOGKB-NcmCacQkeAS75RA02ba...

```

class Enemy {
  -type: String
  +Enemy(Name, level, Enemy)
  +Scream()
  +displayAttributes()
}

class Player {
  -Class: String
  +Player(Name, level, Class)
  +useAbility(enemy)
  +displayAttributes()
}

class Main {
  +main(args)
}

Enemy --|> Character
Player --|> Character
@enduml

```

report this ad

UML Class Diagram:

- Enemy** (Class):
 - Attributes: type: String
 - Operations: Enemy(Name, level, Enemy), Scream(), displayAttributes()
- Player** (Class):
 - Attributes: Class: String
 - Operations: Player(Name, level, Class), useAbility(enemy), displayAttributes()
- Main** (Class):
 - Operations: main(args)
- Character** (Class):
 - Attributes: Name: String, Level: int
 - Operations: Character(Name, level), Attack(level), displayAttributes()

Relationships: Enemy and Player are subclasses of Character.

PlantUML Web Server

https://www.plantuml.com/plantuml/uml/SoWklmgAsiDuNBEImkDZ1KiAdHrLM0S8oWWiOAMd0n4wY0gK8-NCmCacQkeAS75RA02ba...

Trabajo

```
class PCream {
    -cream: Boolean
    +PCream(Flavor, price, cream)
    +showCream()
    +ChangePrice()
}

class PWater {
    -BaseWater: Boolean
    +PWater(Flavor, Price, Base)
    +showBaseW()
    +ChangePrice()
}

class Main {
    +main(args)
}

PCream --> palette
PWater --> palette
@enduml
```

[//www.plantuml.com/plantuml/png/VP6nQID038PtFull6AubVm4C92HGw50mTEgJdn36STK0djgFRTwzFvvfwoIRz_oS-x3ibHT5v39KFc19C64YLua12g9y2VahFu5](#) [Decode URL](#)

Submit

Pure Javascript

PNG SVG ASCII Art

chat 47 online

online diagrams 274,299,066

current rate 261 diag. per minute

peak rate 1004 diag. per minute

PCream

cream: Boolean

PCream(Flavor, price, cream)

showCream()

ChangePrice()

PWater

BaseWater: Boolean

PWater(Flavor, Price, Base)

showBaseW()

ChangePrice()

Main

main(args)

palette

Flavor: String

price: double

palette(Flavor, price)

showInfo()

discount(percentage)

report this ad