

Personas Python

Primero definimos la clase base para representar a una persona, la cual contiene atributos como id, nombre, curp y domicilio, los cuales representan características generales de una persona. Inicializamos los datos con el constructor.

```
1 class Persona:
2     def __init__(self, id, nombre, curp, domicilio):
3         self.id = id
4         self.nombre = nombre
5         self.curp = curp
6         self.domicilio = domicilio
7
```

Aquí definimos un método abstracto para obtener los valores de una persona e indicamos que este método debe ser implementado en cada una de las clases derivadas.

```
8     def obtener_valores(self):
9         pass
```

Después se implementan los métodos que devuelven una cadena de caracteres para mostrar los valores obtenidos de un estudiante, un docente o un PAAE.

```
class Estudiante(Persona):
    def obtener_valores(self):
        return "ID: " + str(self.id) + "\nNombre: " + self.nombre + "\nCURP: " + self.curp + "\nDomicilio: " + self.domicilio

class Docente(Persona):
    def obtener_valores(self):
        return "ID: " + str(self.id) + "\nNombre: " + self.nombre + "\nCURP: " + self.curp + "\nDomicilio: " + self.domicilio

class PAAE(Persona):
    def obtener_valores(self):
        return "ID: " + str(self.id) + "\nNombre: " + self.nombre + "\nCURP: " + self.curp + "\nDomicilio: " + self.domicilio
```

Al final realizamos un menú en el cual podemos registrar y mostrar los datos de un alumno, un docente o un PAAE según la opción seleccionada.

Se solicita al usuario que ingrese el ID de la persona utilizando input y se convierte en un entero utilizando la función int() para asignársela a la variable id.

Se realiza lo mismo para el nombre, CURP y domicilio con la diferencia de que se usa la función str() para convertirlos en cadena de caracteres.

Después se crea una instancia de la persona seleccionada, pasando los valores de id, nombre, curp y domicilio como argumentos al constructor.

Por ultimo se llama al método obtener_valores() en el objeto de la persona seleccionada para mostrar los datos registrados.

```

def menu():
    while True:
        print("----- Menu de Opciones -----")
        print("1. Obtener valores de un estudiante")
        print("2. Obtener valores de un docente")
        print("3. Obtener valores de un PAEE")
        print("4. Salir")
        opcion = input("Seleccione una opción: ")

        if opcion == "1":
            id = int(input("Ingrese el ID del estudiante: "))
            nombre = str(input("Ingrese el nombre del estudiante: "))
            curp = str(input("Ingrese el CURP del estudiante: "))
            domicilio = str(input("Ingrese el domicilio del estudiante: "))
            estudiante = Estudiante(id, nombre, curp, domicilio)
            print("\n" + paae.obtener_valores() + "\nLos datos del estudiante se han registrado correctamente!")

        elif opcion == "2":
            id = int(input("Ingrese el ID del docente: "))
            nombre = str(input("Ingrese el nombre del docente: "))
            curp = str(input("Ingrese el CURP del docente: "))
            domicilio = str(input("Ingrese el domicilio del docente: "))
            docente = Docente(id, nombre, curp, domicilio)
            print("\n" + paae.obtener_valores() + "\nLos datos del docente se han registrado correctamente!")

        elif opcion == "3":
            id = int(input("Ingrese el ID del PAEE: "))
            nombre = str(input("Ingrese el nombre del PAEE: "))
            curp = str(input("Ingrese el CURP del PAEE: "))
            domicilio = str(input("Ingrese el domicilio del PAEE: "))
            paae = PAEE(id, nombre, curp, domicilio)
            print("\n" + paae.obtener_valores() + "\nLos datos del PAEE se han registrado correctamente!")

        elif opcion == "4":
            break

        else:
            print("Opción inválida. Por favor, seleccione una opción válida.")

    print()

# Ejecutar el menú
menu()

```

Personas Java

Primero importamos la clase Scanner del paquete java.util para permitir la entrada de datos desde el teclado, después definimos una clase abstracta llamada Persona la cual contiene atributos como id, nombre, curp y domicilio, los cuales representan características generales de una persona. Además, tiene un constructor que inicializa estos atributos y un método abstracto llamado obtenerValores().

```
import java.util.Scanner;

abstract class Persona {
    public double id;
    public String nombre;
    public String curp;
    public String domicilio;

    public Persona(double id, String nombre, String curp, String domicilio) {
        this.id = id;
        this.nombre = nombre;
        this.curp = curp;
        this.domicilio = domicilio;
    }

    public abstract String obtenerValores();
}
```

Se definen las clases Estudiante, Docente y PAAE, que heredan de la clase Persona. Estas clases implementan el método obtenerValores() que devuelve una cadena de caracteres para mostrar los valores obtenidos de un estudiante, un docente o un PAAE.

```
class Estudiante extends Persona {
    public Estudiante(double id, String nombre, String curp, String domicilio) {
        super(id, nombre, curp, domicilio);
    }

    public String obtenerValores() {
        return "ID: " + id + "\nNombre: " + nombre + "\nCURP: " + curp + "\nDomicilio: " + domicilio;
    }
}

class Docente extends Persona {
    public Docente(double id, String nombre, String curp, String domicilio) {
        super(id, nombre, curp, domicilio);
    }

    public String obtenerValores() {
        return "ID: " + id + "\nNombre: " + nombre + "\nCURP: " + curp + "\nDomicilio: " + domicilio;
    }
}

class PAAE extends Persona {
    public PAAE(double id, String nombre, String curp, String domicilio) {
        super(id, nombre, curp, domicilio);
    }

    public String obtenerValores() {
        return "ID: " + id + "\nNombre: " + nombre + "\nCURP: " + curp + "\nDomicilio: " + domicilio;
    }
}
```

Al final realizamos un menú en el cual podemos registrar y mostrar los datos de un alumno, un docente o un PAAE según la opción seleccionada utilizando Scanner

Después se crea una instancia de la persona seleccionada, pasando los valores de id, nombre, curp y domicilio como argumentos al constructor.

Por último se llama al método obtener valores() en el objeto de la persona seleccionada para mostrar los datos registrados.

```
public class Personas {
    Run | Debug
    public static void main(String[] args) {
        Scanner scanner = new Scanner(System.in);

        while (true) {
            System.out.println(x: "----- Menu de Opciones -----");
            System.out.println(x: "1. Obtener valores de un estudiante");
            System.out.println(x: "2. Obtener valores de un docente");
            System.out.println(x: "3. Obtener valores de un PAAE");
            System.out.println(x: "4. Salir");
            System.out.print(s: "Seleccione una opción: ");
            String opcion = scanner.nextLine();

            if (opcion.equals(anObject: "1")) {
                System.out.print(s: "Ingrese el ID del estudiante: ");
                double id = scanner.nextDouble();
                scanner.nextLine();
                System.out.print(s: "Ingrese el nombre del estudiante: ");
                String nombre = scanner.nextLine();
                System.out.print(s: "Ingrese el CURP del estudiante: ");
                String curp = scanner.nextLine();
                System.out.print(s: "Ingrese el domicilio del estudiante: ");
                String domicilio = scanner.nextLine();

                Estudiante estudiante = new Estudiante(id, nombre, curp, domicilio);
                System.out.println(estudiante.obtenerValores());
            } else if (opcion.equals(anObject: "2")) {
                System.out.print(s: "Ingrese el ID del docente: ");
                double id = scanner.nextDouble();
                scanner.nextLine();
                System.out.print(s: "Ingrese el nombre del docente: ");
                String nombre = scanner.nextLine();
                System.out.print(s: "Ingrese el CURP del docente: ");
                String curp = scanner.nextLine();
                System.out.print(s: "Ingrese el domicilio del docente: ");
                String domicilio = scanner.nextLine();

                Docente docente = new Docente(id, nombre, curp, domicilio);
                System.out.println(docente.obtenerValores());
            }
        }
    }
}
```

```
} else if (opcion.equals(anObject:"3")) {
    System.out.print(s:"Ingrese el ID del PAEE: ");
    double id = scanner.nextDouble();
    scanner.nextLine();
    System.out.print(s:"Ingrese el nombre del PAEE: ");
    String nombre = scanner.nextLine();
    System.out.print(s:"Ingrese el CURP del PAEE: ");
    String curp = scanner.nextLine();
    System.out.print(s:"Ingrese el domicilio del PAEE: ");
    String domicilio = scanner.nextLine();

    PAEE paae = new PAEE(id, nombre, curp, domicilio);
    System.out.println(paae.obtenerValores());

} else if (opcion.equals(anObject:"4")) {
    break;

} else {
    System.out.println(x:"Opción inválida. Por favor, seleccione una opción válida.");
}

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

ÁREA Y PERIMETRO DE FIGURAS GEOMETRICAS JAVA

Se define una clase abstracta llamada FiguraGeometrica que tiene dos métodos abstractos: calcularArea() y calcularPerimetro(). Estos métodos serán implementados en las clases derivadas.

```
abstract class FiguraGeometrica {  
    public abstract double calcularArea();  
    public abstract double calcularPerimetro();  
}
```

Se define la clase Triangulo que hereda de FiguraGeometrica. Tiene dos atributos: base y altura. Se implementan los métodos abstractos calcularArea() y calcularPerimetro() de acuerdo a la fórmula correspondiente al triángulo.

```
class Triangulo extends FiguraGeometrica {  
    private double base;  
    private double altura;  
  
    public Triangulo(double base, double altura) {  
        this.base = base;  
        this.altura = altura;  
    }  
  
    public double calcularArea() {  
        return (base * altura) / 2;  
    }  
  
    public double calcularPerimetro() {  
        return 3 * base;  
    }  
}
```

Se hace lo mismo para el círculo, rectángulo y hexágono.

```
class Circulo extends FiguraGeometrica {  
    private double radio;  
  
    public Circulo(double radio) {  
        this.radio = radio;  
    }  
  
    public double calcularArea() {  
        return Math.PI * Math.pow(radio, 2);  
    }  
  
    public double calcularPerimetro() {  
        return 2 * Math.PI * radio;  
    }  
}
```

```
class Rectangulo extends FiguraGeometrica {
    private double ladoA;
    private double ladoB;

    public Rectangulo(double ladoA, double ladoB) {
        this.ladoA = ladoA;
        this.ladoB = ladoB;
    }

    public double calcularArea() {
        return ladoA * ladoB;
    }

    public double calcularPerimetro() {
        return 2 * (ladoA + ladoB);
    }
}
```

```
class Hexagono extends FiguraGeometrica {
    private double lado;

    public Hexagono(double lado) {
        this.lado = lado;
    }

    public double calcularArea() {
        return (Math.sqrt((lado*lado)-((lado/2)*(lado/2)))*(lado)*6)/2;
    }

    public double calcularPerimetro() {
        return 6 * lado;
    }
}
```

Se define la clase Figuras que contiene el método main(). Se crea un objeto Scanner para leer la entrada del usuario.

```
public class Figuras {
    Run | Debug
    public static void main(String[] args) {
        Scanner scanner = new Scanner(System.in);
```

Se muestra un menú de opciones al usuario y se lee la opción seleccionada. Dependiendo de la opción elegida, se ejecuta el código correspondiente para calcular el área y perímetro de la figura geométrica seleccionada. Si la opción es

"5", se sale del bucle while. Si la opción no es válida, se muestra un mensaje de error.

```
while (true) {
    System.out.println(x:"----- Menú de Opciones -----");
    System.out.println(x:"1. Calcular área y perímetro de un triángulo");
    System.out.println(x:"2. Calcular área y perímetro de un círculo");
    System.out.println(x:"3. Calcular área y perímetro de un rectángulo");
    System.out.println(x:"4. Calcular área y perímetro de un hexágono");
    System.out.println(x:"5. Salir");
    System.out.print(s:"Seleccione una opción: ");

    String opcion = scanner.nextLine();

    if (opcion.equals("1")) {
        // Código para calcular el área y perímetro de un triángulo
    } else if (opcion.equals("2")) {
        // Código para calcular el área y perímetro de un círculo
    } else if (opcion.equals("3")) {
        // Código para calcular el área y perímetro de un rectángulo
    } else if (opcion.equals("4")) {
        // Código para calcular el área y perímetro de un hexágono
    } else if (opcion.equals("5")) {
        break;
    } else {
        System.out.println(x:"Opción inválida. Por favor, seleccione una opción válida.");
    }

    System.out.println();
    scanner.nextLine(); // Consumir el salto de línea después de leer un valor numérico
}
```

PHYTON

Se importan los módulos necesarios. ABC y abstractmethod son utilizados para definir una clase y métodos abstractos, respectivamente. Math se utiliza para acceder a funciones matemáticas, como el valor de pi y la función de raíz cuadrada.

```
from abc import ABC, abstractmethod
import math
```

Se definen las clases que representan diferentes figuras geométricas. La clase FiguraGeometrica es una clase abstracta que define los métodos abstractos calcular_area y calcular_perimetro.

```
class FiguraGeometrica(ABC):  
    @abstractmethod  
    def calcular_area(self):  
        pass  
  
    @abstractmethod  
    def calcular_perimetro(self):  
        pass
```

Las clases Triangulo, Circulo, Rectangulo y Hexagono heredan de FiguraGeometrica y proporcionan implementaciones concretas de los métodos abstractos.

```
class Triangulo(FiguraGeometrica):  
    def __init__(self, base, altura):  
        self.base = base  
        self.altura = altura  
  
    def calcular_area(self):  
        return (self.base * self.altura) / 2  
  
    def calcular_perimetro(self):  
        return 3 * self.base  
  
class Circulo(FiguraGeometrica):  
    def __init__(self, radio):  
        self.radio = radio  
  
    def calcular_area(self):  
        return math.pi * (self.radio ** 2)  
  
    def calcular_perimetro(self):  
        return 2 * math.pi * self.radio  
  
class Rectangulo(FiguraGeometrica):  
    def __init__(self, lado_a, lado_b):  
        self.lado_a = lado_a  
        self.lado_b = lado_b  
  
    def calcular_area(self):  
        return self.lado_a * self.lado_b  
  
    def calcular_perimetro(self):  
        return 2 * (self.lado_a + self.lado_b)
```

```

class Hexagono(FiguraGeometrica):
    def __init__(self, lado):
        self.lado = lado

    def calcular_area(self):
        return (math.sqrt((self.lado*self.lado)-((self.lado/2)*(self.lado/2))

    def calcular_perimetro(self):
        return 6 * self.lado

```

Este bloque define la función menu que muestra el menú de opciones al usuario y realiza los cálculos correspondientes según la opción seleccionada. Luego, se ejecuta la función menu para iniciar el programa. El bucle while True asegura que el menú se ejecute continuamente hasta que el usuario elija la opción de salida.

```

def menu():
    while True:
        print("----- Menú de Opciones -----")
        print("1. Calcular área y perímetro de un triángulo")
        print("2. Calcular área y perímetro de un círculo")
        print("3. Calcular área y perímetro de un rectángulo")
        print("4. Calcular área y perímetro de un hexagono")
        print("5. Salir")
        opcion = input("Seleccione una opción: ")

```

Dentro del bucle, se muestra el menú y se solicita al usuario que ingrese una opción. Luego, se utilizan estructuras if-elif-else para determinar qué cálculos realizar según la opción seleccionada.

Para cada opción, se solicitan los valores necesarios para realizar los cálculos y se crean instancias de las clases correspondientes (Triangulo, Circulo, Rectangulo o Hexagono). Luego, se llaman a los métodos calcular_area y calcular_perimetro de la instancia creada y se imprimen los resultados.

```
if opcion == "1":
    base = float(input("Ingrese la base del triángulo: "))
    altura = float(input("Ingrese la altura del triángulo: "))
    triangulo = Triangulo(base, altura)
    print("Área:", triangulo.calcular_area())
    print("Perímetro:", triangulo.calcular_perimetro())

elif opcion == "2":
    radio = float(input("Ingrese el radio del círculo: "))
    circulo = Circulo(radio)
    print("Área:", circulo.calcular_area())
    print("Perímetro:", circulo.calcular_perimetro())

elif opcion == "3":
    lado_a = float(input("Ingrese el lado A del rectángulo: "))
    lado_b = float(input("Ingrese el lado B del rectángulo: "))
    rectangulo = Rectangulo(lado_a, lado_b)
    print("Área:", rectangulo.calcular_area())
    print("Perímetro:", rectangulo.calcular_perimetro())

elif opcion == "4":
    lado = float(input("Ingrese el lado del hexagono: "))
    hexagono = Hexagono(lado)
    print("Área:", hexagono.calcular_area())
    print("Perímetro:", hexagono.calcular_perimetro())

elif opcion == "5":
    break

else:
    print("Opción inválida. Por favor, seleccione una opción válida.")

print()
```

```
# Ejecutar el menú
menu()
```