Creación de objetos

Dr. José Lázaro Martínez Rodríguez

Creación de objetos (Instancias)

- El proceso de crear objetos de una clase se llama instanciación.
- Los objetos se manipulan con referencias.
- Una referencia es una variable que apunta a un objeto.
- Los objetos se crean (se instancian) con el operador de instanciación new
- Ejemplo:

```
Punto p;
p = new Punto();
//sería lo mismo Punto p = new Punto();
```

Creación de objetos (Instancias)

• Declaremos que los objetos e y f son de la clase Ejemplo:

```
Ejemplo e;
Ejemplo f;
```

- Declarar las variables e y f de la clase Ejemplo no implica que se crearon los objetos.
- En realidad, hasta aquí, sólo se crean apuntadores a objetos de esta clase. Para crear un objeto, se usa el operador new.

```
Ejemplo e;

Ejemplo f;

e = new Ejemplo();

f = new Ejemplo();
```

Creación de objetos (Instancias)

• En general, la manera de crear un objeto de una clase X es la siguiente:

```
// primero declarar el objeto indicando su clase
<Clase> <nombreObjeto>;
// después crear el objeto
<nombreObjeto> = new <Clase>();
```

• O con un solo enunciado

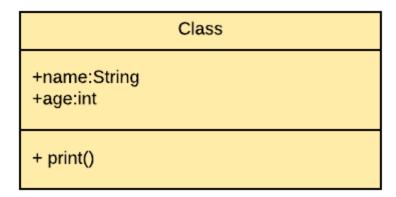
```
<Clase> <nombreObjeto> = new <Clase>();
```

Acceso

- Para acceder a los miembros de un objeto se hace a través de su referencia y el operador.
- Nombre referencia.miembro
- Ejemplo:

```
p.x = 1; //acceder a datos miembro
p.y = 3;
//acceder a métodos miembro
p.asignarCoordenadas();
```

- Escribe un programa Java para crear una clase llamada "Persona" con un nombre y un atributo de edad.
- Cree dos instancias de la clase "Persona.
- Imprimir el nombre y la edad



• Escribe un programa Java para crear una clase llamada "Rectángulo" con atributos de anchura y altura. Calcule el área y el perímetro del rectángulo.

```
public class Rectangulo {
double anchura;
double altura;
}
```

• Podemos definir métodos para asignar valores

```
public class Rectangulo {
    double anchura;
    double altura;

public void definirAnchura(double nuevaAnchura) {
        anchura = nuevaAnchura;
}

public void definirAltura(double nuevaAltura) {
        anchura = nuevaAltura;
}
```

• Definimos métodos para obtener área y perímetro

```
public double obtenerArea() {
    return anchura * altura;
}

public double obtenerPerimetro() {
    return anchura * 2 + altura * 2;
}
```

• Ahora creamos instancias dentro del main

```
public static void main(String[] args) {
          Rectangulo r1 = new Rectangulo();
          r1.definirAltura(10);
          r1.definirAnchura(5);

double areaR1 = r1.obtenerArea();

System.out.println("Area de r1: " + areaR1);

System.out.println("Perímetro de r1: " + r1.obtenerPerimetro());
}
```

Ejercicio

- Declarar la clase Cuenta con los atributos nombre, saldo, numero y tipo, y con los métodos depositar, que recibe como parámetro la cantidad a depositar, y retirar, que recibe como parámetro la cantidad a retirar. Sólo se efectuará el retiro si el saldo es mayor o igual a la cantidad a retirar.
- Escribir un método que imprima los datos del objeto.

private String nombre; private double saldo; private int numero; private String tipo; public void depositar(double deposito) {} public void retirar (double retiro) {} publicvoid imprimir () {}

```
public class Cuenta {
         String nombre;
         double saldo=0; //empieza en cero
         int numero;
         String tipo;
}
```

Métodos

• Y para imprimir

```
public void imprimir() {
   System.out.println("Datos de la cuenta");
   System.out.println("Nombre " + nombre);
   System.out.println("Saldo " + saldo);
   System.out.println("Numero " + numero);
   System.out.println("Tipo " + tipos);
}
```

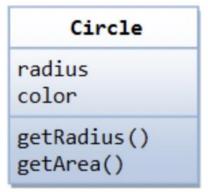
Ejercicio

- Crear los objetos cuentaCredito y cuentaDebito en la clase Principal.
- Al objeto cuentaCredito ponerlo a nombre de "Pedro Sánchez", con saldo de 1500, con número de cuenta 244513, indicando que es una cuenta de crédito.
- Al objeto cuentaDebito ponerlo a nombre de Pablo García, con saldo de 7800, con número de cuenta 273516, indicando que es una cuenta de débito.
- ¿Qué es lo que se imprime?

Ejercicio

• Crear las siguientes clases en java

Student name grade getName() printGrade()



 Además, crear diez instancias de cada clase con sus respectivos valores

```
name
number
xLocation
yLocation
run()
jump()
kickBall()
```

```
plateNumber
xLocation
yLocation
speed
move()
park()
accelerate()
```