

Java • hoja de referencia: POO

CLASE CON ATRIBUTOS Y CONSTRUCTOR

```
class Estudiante {
    private String nombre;
    private int legajo;

    public Persona(String nombre, int nLeg)
    { //código }
}
```

INSTANCIAR OBJETO

DECLARACIÓN

```
Estudiante juan;
```

INSTANCIACIÓN

```
juan=new Estudiante("juan", "52245");
```

MODIFICADOR STATIC

ATRIBUTOS DE CLASE

```
static String escuela;
```

MÉTODOS DE CLASE

```
static void darNombreAEscuela(String n) {
    Estudiante.escuela=n; }
}
```

INVOCAR MÉTODO DE CLASE

```
Estudiante.darNombreAEscuela("ABC");
```

- no se requiere una instancia del objeto para utilizarlos.
- un método static no puede usar atributos no static.

MODIFICADOR ABSTRACT

CLASE ABSTRACT

```
public abstract class Personal { }
```

- no pueden ser instanciadas (sólo heredadas).

MÉTODOS ABSTRACT

```
public abstract double calcularSalario();
```

- no pueden ser implementados (lo hace la clase hija).

INTERFACES

```
public interface Empleado {
    public long calcularAntiguedad(); }
```

- sólo permiten el modificador public.



Programación
Desde Cero

MODIFICADORES DE ACCESO

public clase, paquete, subclase, resto del programa
protected clase, paquete, subclase
sin modificador clase, paquete
private clase

GETTERS Y SETTERS PARA LOS ATRIBUTOS

GETTER

```
public int getLegajo(){
    return this.legajo;
}
```

SETTER

```
public void setLegajo(int nLeg){
    this.legajo=legajo;
}
```

MODIFICADOR FINAL

ATRIBUTOS FINAL

```
private final long dni = 39243612;
```

- su contenido no puede variar. Si es un objeto, no puede cambiar la referencia, pero sí el estado del objeto referenciado.

MÉTODOS FINAL

```
private final calcularPromedio() { }
```

- no puede ser sobrescrito por las subclases.

CLASES FINAL

```
public final class Asignatura { }
```

- no pueden ser heredadas (pero sí instanciadas).

COLECCIONES

Contenedor	Orden de los elementos	Acceso aleatorio	Pares clave-valor	Permite duplicados	Permite valores nulos
HashSet	✗	✓	✗	✗	✓
TreeSet	✓	✓	✗	✗	✗
LinkedHashSet	✓	✗	✗	✗	✓
ArrayList	✓	✓	✗	✓	✓
LinkedList	✓	✗	✗	✓	✓
Vector	✓	✓	✗	✓	✓
LinkedBlockingQueue	✓	✗	✗	✓	✗
ArrayBlockingQueue	✓	✗	✗	✓	✗
PriorityBlockingQueue	✓	✗	✗	✓	✗
DelayQueue	✓	✗	✗	✓	✗
SynchronousQueue	✓	✗	✗	✓	✗
HashMap	✗	✓	✓	✗	✓
TreeMap	✓	✓	✓	✗	✗
LinkedHashMap	✓	✗	✓	✗	✓