

# Tarea presencial.

## Unidad 5. Desarrollo de clases

---

### Indicaciones de entrega

Una vez realizada la tarea, el envío se realizará a través de la plataforma Moodle, en la tarea "Tarea presencial 4. Desarrollo de clases POO".

Debes comprimir la carpeta que contiene el proyecto en un archivo. Este archivo comprimido se nombrará de la siguiente forma:

*Apellido1\_Apellido2\_Nombre\_Tarea4.zip*

### Evaluación

#### Cómo se valora y puntúa la tarea

Estructuras herencias e implementaciones → 2.5 puntos

Métodos cálculo de sueldos → 1 punto

Método contratar empleado → 0.75 puntos

Método para obtener empleados por nombre → 1 punto

Métodos para mostrar los datos → 2 puntos

Control de errores → 1 punto

Resultado de uso correcto → 0.75 puntos

#### Factores que influyen en la puntuación de cada ejercicio

- Principal: El código debe compilar y ejecutarse correctamente
- Errores de base sobre conceptos de temas anteriores: nombres de identificadores incorrectos, elección de tipo inadecuado, etc.
- Elección de la estructura de control de flujo más apropiada para cada solución: if-else, switch-case, for, while...
- Calidad, legibilidad y limpieza del código.
- Control y manejo de excepciones que proporcionen una buena experiencia de usuario.
- La implementación de métodos que modularicen los programas.

#### Resultados de aprendizaje que se trabajan en la tarea

**RA4:** Desarrolla programas organizados en clases analizando y aplicando los principios de la programación orientada a objetos.

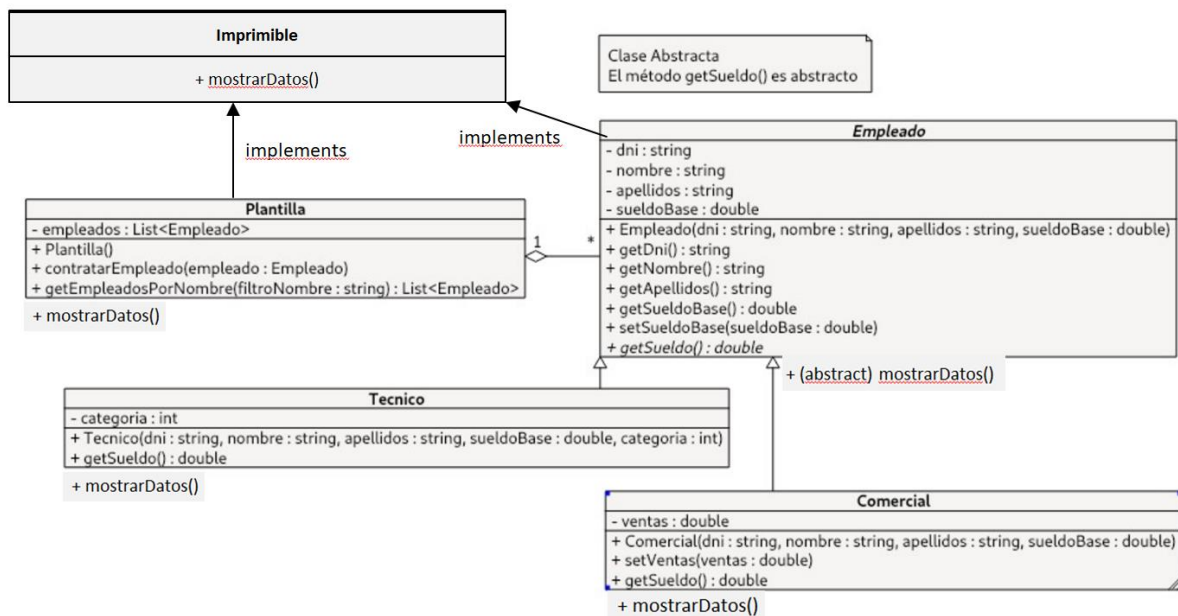
**RA7:** Desarrolla programas aplicando características avanzadas de los lenguajes orientados a objetos y del entorno de programación.

## Enunciado

Observa el siguiente diagrama de clases:

En el diagrama se describe la estructura para representar la plantilla de empleados de una empresa. Los empleados se dividen en técnicos y comerciales. El sueldo de los técnicos dependerá de su categoría (número entero) y el de los comerciales, de las ventas realizadas.

Desarrolla la estructura representada en el diagrama.



Ten en cuenta las siguientes consideraciones:

- En la clase Tecnico, el sueldo se calculará como el sueldo base más la categoría multiplicada por 100.
- En la clase Comercial, el sueldo se calculará como el sueldo base más un 10% de las ventas realizadas.
- En la clase Plantilla, el método contratarEmpleado() añadirá el empleado dado a la plantilla.
- Nombre, apellidos y DNI no pueden estar vacíos. El sueldo base no puede ser menor del SMI 1.134 euros.
- El método getEmpleadosPorNombre() devolverá una lista con todos los empleados que contengan en el nombre, o en los apellidos, el texto pasado como parámetro.
- Las clases Empleado y Plantilla implementan a la interfaz Imprimible. Este método imprimirá por consola los valores de los objetos de dichos tipos.
  - En la clase Empleado es abstracto y deberá implementarse en las clases Tecnico y Comercial.
  - En la clase Plantilla, este método usará los métodos homónimos de las clases Técnico y comercial.

- El código debe cumplir, estrictamente, esta estructura:

```
public XXXXXX class Empleado XXXXXX {  
    // código de la clase Empleado  
}
```

```
public XXXXXX class Tecnico XXXXXX {  
    // código de la clase Tecnico  
}
```

```
public XXXXXX class Comercial XXXXXX {  
    // código de la clase Comercial  
}
```

```
import java.util.*;  
public XXXXXX class Plantilla XXXXXX {  
    // código de la clase Plantilla  
}
```

(XXXXXX representa texto que debe completar, o dejar vacío, dependiendo del caso).

- No se debe añadir ninguna funcionalidad no solicitada.
- No se debe añadir ninguna librería, no se debe especificar más imports. Todas las clases están en el mismo paquete.
- Respeta los nombres que aparecen en el diagrama.
- Se valorará positivamente la limpieza, simplicidad y la claridad en el código.
- Se valorará el uso de comentarios allá donde sea conveniente.

A continuación puedes ver un ejemplo aclaratorio del uso de las clases descritas, junto con la salida generada:

#### Ejemplo de Uso:

```
public static void main(String[] args) {  
    Plantilla plantillaTrafalgar = new Plantilla();  
    // Crear y contratar un técnico de categoría 1 llamado Alejandro Fernández  
    //con dni 11111111H y con 1000 de sueldo base  
    plantilla.contratarEmpleado(new Tecnico("11111111H", "Alejandro",  
    "Fernández", 1000, 1));  
  
    // Crear y contratar una técnico de categoría 2 llamada Jerónima Jiménez con  
    //dni 22222222J y con 1000 de sueldo base  
    plantilla.contratarEmpleado(new Tecnico("22222222J", "Jerónima", "Jiménez",  
    1000, 2));  
  
    // Crear y contratar un comercial llamado Desiderio Durán con dni 33333333P  
    // y con 800 de sueldo base. Establecer sus ventas a 2000  
    Comercial comercial = new Comercial(33333333P, "Desiderio", "Durán", 800);  
    comercial.setVentas(2000);  
    Plantilla.contratarEmpleado(comercial);  
    // Muestra todos los empleados de la plantilla "plantillaTrafalgar"  
    plantilla.mostrarDatos();  
}
```

#### Resultado:

```
11****11H - Alejandro Fernández: 1100 €  
22****22J - Jerónima Jiménez: 1200 €  
33****33P - Desiderio Durán: 1000 €
```