

Semana 3

Desarrollo Orientado a Objetos I (PRY2202)

# Descripción de la actividad

En esta tercera semana realizarás una actividad sumativa individual con encargo llamada "Estructurando paquetes y directorios con uso de clases", donde, a través del caso planteado en semanas anteriores y usar el proyecto creado e implementar clases y los atributos de cada una de ellas. Además, deberás generar los constructores, accesadores y mutadores y los métodos necesarios que cumplan con los requerimientos y reglas de negocio solicitadas. Además, deberás utilizar herencia y aplicar superclase, subclases, modificación de métodos y clases anexas.

## Instrucciones específicas

Para realizar la actividad sumativa, revisa el caso planteado:

Bank Europe se encuentra en proceso de crecimiento y posicionamiento, tanto a nivel operacional como estructural; sus servicios ofrecidos se encuentran en continuas etapas de perfeccionamiento, como por ejemplo, su servicio de gestión de cuentas bancarias, donde:

1. Cada cuenta tiene un número de cuenta de 9 dígitos y saldo.
2. Cada cliente puede tener una sola cuenta contratada, y realizar operaciones como registrar y ver sus datos, depositar y girar dinero y consultar saldo.
3. Cada cliente tiene sus características personales, y
4. Toda cuenta corriente tiene las siguientes características:

* Número: Ej: 123456789
* Saldo: Ej: 1550900

A partir de aquí, deberás aplicar lo aprendido en las semanas 1 y 2 y así poder entregar una solución final al caso planteado. Para ello, tendrás que seguir los siguientes pasos:

**Paso 1: Creación de la clase base abstracta**

* Crea un nuevo proyecto en Java llamado "Bank\_Europe".
* Dentro del proyecto, crea una clase base abstracta llamada ‘CuentaBancaria’.
* En la clase ‘CuentaBancaria’, define al menos dos atributos como saldo y número de cuenta, encapsulándolos adecuadamente.
* Crea un constructor en la clase ‘CuentaBancaria’ para inicializar estos atributos.
* Declara un método abstracto llamado ‘calcularInteres’ que representará el cálculo de intereses. Esta será una característica común a todas las cuentas bancarias.

**Paso 2: Creación de subclases**

* Crea tres subclases: ‘CuentaCorriente’, ‘CuentaAhorros’ y ‘CuentaDigital’. Todas estas subclases heredarán de la clase ‘CuentaBancaria’.
* En cada subclase, implementa el método abstracto ‘calcularInteres’ de acuerdo con el tipo de cuenta, ya que el cálculo de intereses puede variar.
* Puedes aplicar la sobrecarga de métodos en la clase base para tener varios constructores con diferentes parámetros, lo que permitirá crear subclases de diferentes maneras.

**Paso 3: Implementación de la interfaz**

* Crea una interfaz llamada ‘InfoCliente’ que defina un método ‘mostrarInformacionCliente’.
* Las clases que necesiten mostrar información del cliente deberán implementar esta interfaz y proporcionar su propia implementación del método según sus necesidades.

**Paso 4: Sobreescritura en la clase abstracta**

* En la clase abstracta ‘CuentaBancaria’, declara el método abstracto ‘calcularInteres’.
* Cada subclase, como ‘CuentaCorriente’, ‘CuentaAhorros’ y ‘CuentaDigital’, deberá sobrescribir este método para adaptarlo a su comportamiento específico.

En el siguiente paso, deberás aplicar lo aprendido en esta semana, es decir:

**Paso 5: Aplicación de jerarquía de paquetes y dominios**

Para realizar este paso, tendrás que realizar lo siguiente:

* Identifica al menos dos dominios en tu proyecto.
* Crea paquetes principales para cada uno de los dominios.
* Organiza las clases con cada dominio en su respectivo paquete.
* Crea dentro de un dominio dos paquetes internos.
* Utiliza importaciones adecuadas en tus clases para acceder a clases de otros paquetes (‘import’).

**Paso 6:** para realizar esta actividad, tendrás que utilizar el entorno de desarrollo Apache NetBeans. Si no lo has descargado, puedes hacerlo desde el siguiente enlace:

<https://netbeans.apache.org/front/main/download/archive/>

**Paso 7:** al finalizar, deberás entregar la solución en formato .zip o .rar.



Reservados todos los derechos Fundación Instituto Profesional Duoc UC. No se permite copiar, reproducir, reeditar, descargar, publicar, emitir, difundir, de forma total o parcial la presente obra, ni su incorporación a un sistema informático, ni su transmisión en cualquier forma o por cualquier medio (electrónico, mecánico, fotocopia, grabación u otros) sin autorización previa y por escrito de Fundación Instituto Profesional Duoc UC La infracción de dichos derechos puede constituir un delito contra la propiedad intelectual.