

## Ejercicios con clases

### Ejercicio 1

Diseña una clase **Cuenta** que represente una cuenta bancaria (número de cuenta y saldo) y permita realizar operaciones como ingresar y retirar una cantidad de dinero, así como realizar una transferencia de una cuenta a otra.

- Represente gráficamente la clase utilizando la notación **UML**
- Defina la **clase** utilizando la sintaxis de Java, definiendo las **variables de instancia** y **métodos** que crea necesarios.
- Implementa cada uno de los métodos de la clase. Los métodos deben actualizar el estado de las variables de instancia y mostrar un mensaje en el que se indique que la operación se ha realizado con éxito.
- Cree un programa en Java (en una clase llamada **CuentaTest**) que realice las siguientes operaciones:
  - Instancia 2 objetos de la clase Cuenta.
  - Ingresar una cantidad en una cuenta y verifica que el saldo se ha incrementado.
  - Reintegra una cantidad en una cuenta y verifica que el saldo ha disminuido.
  - Transfiere una cantidad de una cuenta a la otra y verifica que los saldos finales sean correctos.
  - Para consultar el saldo de las cuentas puedes implementar un método **toString()**.

### Ejercicio 2

Crea una clase con métodos para cada una de las siguientes opciones del menú:

1. Suma matrices
2. Resta de matrices
3. Producto escalar (el producto escalar da como resultado una matriz cuyos elementos han sido multiplicados por un numero)
0. Salir

- Los atributos de la clase son dos matrices cuadradas de 4x4 que se inicializan con números aleatorios de 0 a 25 (en el constructor).

- Implementa los métodos que permitan acceder a cada uno de los atributos de la clase. Los nombres de dichos métodos serán: **getXXX ()** para obtener el valor del atributo XXX.
- Crea un método estático de nombre **pasoAString** para convertir una matriz cualquiera en un String. Puedes usar este método para mostrar las matrices por pantalla.
- El método **productoEscalar** recibe dos parámetros: el número por el que multiplicar la matriz y un segundo número que indica qué matriz de las dos hay que multiplicar.
- El **menú** se mostrará al usuario hasta que elija una de las opciones (se realizará la acción y el programa finalizará) o introduzca 0 (mostrará un mensaje de despedida) finalizando el programa.

## Ejercicio 3

Construye un nuevo proyecto Java denominado **Ud5\_EjercClases3**. En el proyecto debe aparecer un paquete, que no puede ser el paquete por defecto, ponle el nombre que creas oportuno. Dentro de dicho paquete:

- Crea una clase denominada **Fecha**. Esta clase no debe contener método main.
- Declara en el fichero de la clase un **tipo enumerado**, denominado **enumMes**, para los meses del año.
- La clase debe contener un atributo para el día, otro para mes (del tipo enumerado declarado) y un tercero para el año.
- Implementa un constructor que inicialice el mes al valor recibido por parámetro y los demás atributos a 0. Observa su cabecera en el siguiente código:

```
Fecha (enumMes mes){
```

- Declara otro constructor que inicialice todos los atributos de la clase. Su cabecera podría ser la siguiente:

```
Fecha (int dia, enumMes mes, int anio){
```

- Implementa los métodos que permitan acceder y modificar cada uno de los atributos de la clase. Los nombres de dichos métodos serán: **getXXX ()** para obtener el valor del atributo XXX y **setXXX (v)** para actualizar el atributo XXX con el valor v. Puedes observar la cabecera de algún método en el siguiente código:

```
public int getDia() {  
  
}
```

```
public void setDia(int dia) {  
  
}
```

- Implementa un método que devuelva **true** si el valor contenido en la fecha es verano y false en caso contrario. Observa su cabecera en el siguiente código:

```
public boolean isSummer(){
```

- Implementa un método que devuelva la diferencia en años con respecto a otra fecha pasada como argumento. Este método se podría usar para calcular, por ejemplo, la edad de una persona. Su cabecera sería la siguiente:

```
public int diferenciaEnAnios(Fecha fechaActual){
```

- Implementa un método que devuelva una cadena con la fecha en formato largo, por ejemplo, 15 de julio de 2020. Observa su cabecera:

```
public String toString (){
```

Ya tenemos nuestra clase **Fecha** implementada. Ahora vamos a probar su funcionalidad desde otra clase, denominada **Principal**, que contendrá el método **main**. Esta clase la debes crear en el mismo paquete que la clase Fecha. Dentro de dicha clase:

- Instancia un objeto de la clase Fecha denominado **objFecha1** con el primer constructor.
- Actualiza los atributos día y año para dicho objeto.
- Muestra la fecha por pantalla en formato largo.
- Muestra un mensaje por pantalla indicando si la fecha es verano.
- Muestra la diferencia en años respecto a la fecha actual.
- Instancia otro objeto de la clase **Fecha** denominado **objFecha2** con el segundo constructor.
- Muestra el año de esta fecha por pantalla.
- Muestra la fecha en formato largo por pantalla.
- Muestra un mensaje por pantalla indicando si la fecha es verano o no.
- Muestra la diferencia en años respecto a la fecha actual.

Observa un ejemplo de ejecución:

Primera fecha, inicializada con el primer constructor

La fecha es: 20 de febrero del 2000

No es verano

Hay 3 años de diferencia respecto al 21 de febrero de 2004

Segunda fecha, inicializada con el segundo constructor

La fecha 2 contiene el año 2015

La fecha es: 15 de julio del 2015

Es verano

Hay 8 años de diferencia respecto al 21 de febrero de 2024

## Ejercicio 4

Crea un paquete en java llamado **ficha\_personal**. Dentro de dicho paquete, crea una clase denominada **FichaPersonal**. Esta clase:

- No dispone de método main.
- Puede almacenar los valores del nombre, fecha de nacimiento y edad. La fecha de nacimiento es de tipo **Fecha** (ejercicio anterior) y la edad es un valor que se calcula en el constructor teniendo en cuenta la fecha actual.
- Tiene un constructor por parámetro.
- Posee un método que devuelve un String con los datos del objeto siguiendo este formato:

```
Nombre: Antonio
Fecha nacimiento: 31 de Octubre de 1978
Edad: 46
```

En el mismo paquete, crea una clase llamada **FichaText**. Esta clase tendrá el método main y deberá:

- Instancia un objeto de la clase Fecha. Para ello el programa pide los respectivos valores al usuario por teclado. La fecha de entrada tendrá el formato **dd/mm/aaaa** de manera obligatoria por lo que deberás implementar algún mecanismo para que se cumpla esta restricción.
- Muestra por pantalla el objeto creado.

**Ampliación:** puedes obligar a que la fecha sea correcta

```
Nombre:
Antonio
Fecha de nacimiento (dd/mm/aaaa):
31/10/1978
Nombre: Antonio
Fecha nacimiento: 31 de Octubre de 1978
Edad: 46
```

## Ejercicio 5

Crea un proyecto Java denominado **Ud5\_EjercClases4**. Dentro del proyecto, crea un paquete denominado **com.prog04.figuras**. Dentro de dicho paquete, crea una clase denominada Rectángulo que:

- Declare atributos para la base y la altura de un rectángulo.
- Declare un constructor vacío que inicialice los atributos a 0.
- Declara un constructor que inicialice base y altura.
- Declare los siguientes métodos:
  - Métodos para actualizar y obtener el valor de cada atributo.
  - **float getArea()**: devuelve el área del rectángulo.
  - **String toString()**: devuelve una cadena conteniendo su área y su altura.
  - **boolean isCuadrado()**: devuelve un booleano indicando si el rectángulo es cuadrado.
- Crea otro paquete con el nombre **com.prog04.principal**. Dentro de dicho paquete crea una clase denominada Principal que contenga el método main. En el método main, instancia al menos dos objetos de la clase Rectangulo y comprueba su funcionamiento.