

Detalles de la tarea de esta unidad.

Enunciado.

En esta unidad has visto las características fundamentales de la programación orientada a objetos y estudiado los conceptos más importantes de este modelo de programación. Has aprendido a crear y manipular objetos, qué son los métodos y cómo se definen los parámetros. También has podido conocer cómo se estructura la Biblioteca de Clases de Java, viendo algunas clases importantes, como por ejemplo las que nos permiten realizar la programación de la consola.

Ejercicio 1

- Construye un nuevo proyecto Java denominado **Prog03_Ejerc1**. 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(){
```

No es necesario utilizar estructuras condicionales pues aún no las hemos trabajado. Se puede implementar este método utilizando operadores de comparación.

- 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 (puedes utilizar el operador de comparación ternario. Punto 5.3 de la Unidad 2).
- Instancia otro objeto de la clase `Fecha` denominando `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.

Observa un ejemplo de ejecución:

```
Primera fecha, inicializada con el primer constructor
La fecha es: 20 de febrero del 2000
No es verano
```

```
Segunda fecha, inicializada con el segundo constructor
La fecha 2 contiene el año 2015
La fecha es: 15 de julio del 2015
Es verano
BUILD SUCCESSFUL (total time: 0 seconds)
```

Ejercicio 2

1. Crea un proyecto Java denominado **Prog03_Ejerc2**.
2. Dentro del proyecto, crea un paquete denominado **com.prog03.figuras**.
3. Dentro de dicho paquete, crea una clase denominada **Rectangulo** 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.
4. Crea otro paquete con el nombre **com.prog03.Principal**.
5. Dentro de dicho paquete crea una clase denominada **principal** que contenga el método **main**.
6. En el método **main**, instancia al menos dos objetos de la clase **Rectangulo** y comprueba su funcionamiento.