

Actividad evaluable vacaciones de navidad de Programación.

He creado 4 clases (**Planta**, **Manzano**, **Helecho**, **PlantaMain**), cada una en su respectivo paquete.

1) Súper Clase Planta. Tiene propiedades genéricas de las plantas. Estas son:

```
private String nombreCientifico;
private byte codigo;
private short edad;
private int litrosRiego;
private long numhojas;
private double diametro;
private float altura;
private boolean conFlor;
```

He creado dos constructores, uno con todas sus propiedades y otro vacío. Se ha encapsulado todas las propiedades creando sus getter y setters correspondientes.

Los métodos definidos son:

- **public void** show() {} => Muestra por consola todas las propiedades de la instancia de un objeto.
- **public** String producción() {} => Este método muestra un String(texto) por defecto, ya que la verdadera funcionalidad se le da al sobrescribirlo en las clases que heredan de Planta (**Subclases: Manzano y Helecho**).
- **public void** nacimientoHojas(**int** hojas) {} => Hace un bucle que simula los meses del año y en cada iteración sumamos el número de hojas que nacen, pasado por parámetro.
- **public void** caidaHojas(**int** hojas) {} => Hace un bucle que simula los meses del año y en cada iteración restamos el número de hojas que se caen, pasado por parámetro.

2) Subclase Manzano. Hereda de la Súper clase Planta, por lo que tiene las propiedades y métodos de esta. Además tiene 2 propiedades específicas de los manzanos:

- **private static int** CantidadM = 0; => Propiedad estática que sirve para llevar la cuenta de cuantos manzanos hay creados.
- **private char** clase;

He creado 3 constructores:

- Constructor que recibe por parámetros todas las propiedades.
- Constructor que recibe por parámetros todas las propiedades excepto 2 de ellas que las automatizamos. Estas son el nombre científico y si tiene flor.
- Constructor vacío.

En todos los constructores hacemos uso de super() para llamar al constructor padre, y usamos la instrucción **cantidadM++** para incrementar la propiedad cada vez que se crea un objeto Manzano.

Los métodos definidos son:

- ```
@Override
```
- **public** String produccion() {} => Se sobrescribe el método producción de Planta, para que dependiendo de la propiedad **clase** devuelva un texto u otro, especificando un rango para la propiedad clase.

- **public static void** cantidadArboles () {} => Método estático que muestra por consola la propiedad **cantidadM**, con formato.
- **public void** nacimientoHojas() {} => Llamamos al método nacimientoHojas del padre (Planta), especificando las hojas que nacen mensualmente.
- **public void** caidaHojas() {} => Llamamos al método caidaHojas del padre (Planta), especificando las hojas que se caen mensualmente.
- **public void** tamano() {} => creamos una variable tamaño en la que almacenamos el valor, con una expresión ternaria. Y mostramos por consola dicha información.

3) **Subclase Helecho.** Hereda de la Súper clase Planta, por lo que tiene las propiedades y métodos de esta. Además tiene 2 propiedades específicas de los helechos:

- **private static int** **CantidadH** = 0; => Propiedad estática que sirve para llevar la cuenta de cuantos helechos hay creados.
- **private int** **esporas**;

He creado 3 constructores:

- Constructor que recibe por parámetros todas las propiedades.
- Constructor que recibe por parámetros todas las propiedades excepto 2 de ellas que las automatizamos. Estas son el nombre científico y si tiene flor.
- Constructor vacío.

En todos los constructores hacemos uso de **super()** para llamar al constructor padre, y usamos la instrucción **cantidadH++** para incrementar la propiedad cada vez que se crea un objeto Manzano

Los métodos definidos son:

- @Override**  
- **public** String produccion() {} => Se sobrescribe el método producción de Planta, para que dependiendo de la propiedad **esporas** devuelva un texto u otro, calificando la producción de esporas en una mata.
- **public static void** cantidadMatas() {} => Método estático que muestra por consola la propiedad **cantidadH**, con formato.
- **public void** nacimientoHojas() {} => Llamamos al método nacimientoHojas del padre (Planta), especificando las hojas que nacen mensualmente.
- **public void** caidaHojas() {} => Llamamos al método caidaHojas del padre (Planta), especificando las hojas que se caen mensualmente.

4) Clase principal PlantaMain.

1. He creado 5 instancias de la clase **Manzano**, usando el constructor preparado con la automatización de 2 propiedades.
2. Uso el método cantidadArboles que muestra por pantalla el número de objetos creados.
3. Mostramos cada uno de los objetos con el método **Show()** + su producción.
4. También usamos el método tamaño que nos dice si el árbol es pequeño o grande.
5. He creado 5 instancias de la clase **Helecho**, usando el constructor preparado con la automatización de 2 propiedades.
6. Uso el método cantidadMatas que muestra por pantalla el número de objetos creados.
7. Mostramos cada uno de los objetos con el método Show() + su producción.

8. Probamos los métodos **nacimientoHojas()** y **caidaHojas()** y posteriormente muestro la actualización de los valores.

Explicación visual de por qué manzanos está en un paquete llamado con semilla y helecho en un paquete llamado sin semillas:

