

3.E. Librerías y paquetes.

Marcar como hecha

1. Librerías de objetos (paquetes).

1.4. Utilización de los paquetes.

Es posible acceder a cualquier clase de cualquier paquete (siempre que ese paquete esté disponible en nuestro sistema, obviamente) mediante la calificación completa de la clase dentro de la estructura jerárquica de paquete. Es decir indicando la trayectoria completa de paquetes desde el paquete raíz hasta la propia clase. Eso se puede hacer utilizando el operador **punto.** para especificar cada subpaquete:

```
paquete_raiz.subpaquete1.subpaquete2. ... .subpaquete_n.NombreClase
```

Por ejemplo:

```
java.lang.String.
```

En este caso se está haciendo referencia a la clase **String** que se encuentra dentro del paquete **java.lang**. Este paquete contiene las clases elementales para poder desarrollar una aplicación Java.

Otro ejemplo podría ser:

```
java.util.regex.Patern.
```

En este otro caso se hace referencia a la clase **Patern** ubicada en el paquete **java.util.regex**, que contiene clases para trabajar con expresiones regulares.

Dado que puede resultar bastante tedioso tener que escribir la trayectoria completa de una clase cada vez que se quiera utilizar, existe la posibilidad de indicar que se desea trabajar con las clases de uno o varios paquetes. De esa manera cuando se vaya a utilizar una clase que pertenezca a uno de esos paquetes no será necesario indicar toda su trayectoria. Para ello se utiliza la sentencia **import (importar):**

```
import paquete_raiz.subpaquete1.subpaquete2. ... .subpaquete_n.NombreClase;
```

De esta manera a partir de ese momento podrá utilizarse directamente **NombreClase** en lugar de toda su trayectoria completa.

Los ejemplos anteriores quedarían entonces:

```
import java.lang.String;
```

```
import java.util.regex.Patern;
```

Si suponemos que vamos a utilizar varias clases de un mismo paquete, en lugar de hacer un **import** de cada una de ellas, podemos utilizar el **comodín** (símbolo **asterisco: "*"**) para indicar que queremos importar todas las clases de ese paquete y no sólo una determinada:

```
import java.lang.*;
```

```
import java.util.regex.*;
```

Si un paquete contiene subpaquetes, el comodín no importará las clases de los subpaquetes, tan solo las que haya en el paquete. La importación de las clases contenidas en los subpaquetes habrá que indicarla explícitamente. Por ejemplo:

```
import java.util.*;
```

```
import java.util.regex.*;
```

En este caso se importarán todas las clases del paquete **java.util** (clases **Date**, **Calendar**, **Timer**, etc.) y de su subpaquete **java.util.regex** (**Matcher** y **Pattern**), pero las de otros subpaquetes como **java.util.concurrent** o **java.util.jar**.

Por último tan solo indicar que en el caso del paquete **java.lang**, no es necesario realizar importación. El compilador, dada la importancia de este

paquete, permite el uso de sus clases sin necesidad de indicar su trayectoria (es como si todo archivo Java incluyera en su primera línea la sentencia `import java.lang.*`).

Autoevaluación

La sentencia `import` nos facilita las cosas a la hora de especificar las clases que queremos utilizar en nuestro archivo Java. Con el uso del comodín (asterisco) podemos importar todas las clases y subpaquetes que se encuentran en un determinado paquete a través de una sola sentencia `import`. ¿Verdadero o falso?

- ☐ Verdadero.
- ☒ Falso.

◀ 3.D. Métodos.

Ir a...

3.F. Entrada y salida de datos por consola. ▶