

## Java Foundations

### Prácticas - Sección 6

#### Problema 1: Validación de un PIN bancario

##### Descripción general

Desarrolle un programa java para validar el PIN bancario de un cliente. Use un bucle `while` para repetir el código hasta que se introduzca un PIN válido.



##### Tarea

1. Declare un PIN entero válido.
2. Solicite al usuario que introduzca el PIN.
3. En un bucle `while`, realice estos pasos:
  - Compare el PIN introducido por el usuario con el PIN ya declarado.
  - Si no se introduce el mismo PIN, pida al usuario que vuelva a introducirlo.
  - Repita el bucle hasta que se introduzca el PIN correcto.
4. Imprima un mensaje que confirme que se ha introducido el PIN correcto y que el usuario ya puede acceder a su cuenta.

Encontrará que el archivo `ValidatePin.java` es un recurso útil para empezar.

## Problema 2: Muestra de los múltiplos de un número

### Descripción general

Desarrolle un programa Java para calcular los múltiplos de un número determinado mediante un bucle `for`.

### Tarea

El usuario debe introducir un número y utilizar un bucle `for` para visualizar todos los múltiplos de ese número del 1 al 12.

Salida esperada:

```
Elija un número: 7
```

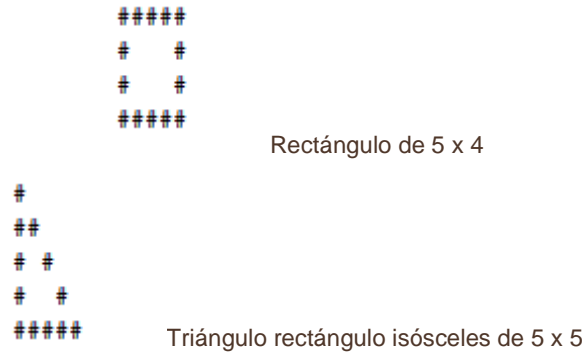
```
7x1 = 7  
7x2 = 14  
7x3 = 21  
7x4 = 28  
7x5 = 35  
7x6 = 42  
7x7 = 49  
7x8 = 56  
7x9 = 63  
7x10 = 70  
7x11 = 77  
7x12 = 84
```

Encontrará que el archivo `DisplayMultiples.java` es un recurso útil para empezar.

## Problema 3: Arte ASCII programático

### Descripción general

Uso de texto para crear una imagen conocida como arte ASCII. En la sección 2 creamos un gato mediante arte ASCII. Para ello, tuvimos que escribir cada carácter con el arte que queríamos crear. En esta práctica encontrará un método para dibujar formas básicas de manera programática en tamaños personalizables.



### Tarea

Complete los dos métodos siguientes en `LoopShape.java`:

- `createRectangle()`: Este método acepta dos argumentos para el ancho y la altura que deben usarse para imprimir un rectángulo
- `createTriangle()`: Este método acepta un argumento para el tamaño de un tramo, que debe usarse para imprimir un triángulo rectángulo isósceles

Intente cambiar el valor de los argumentos que está proporcionando a estos dos métodos desde el método principal. Asegúrese de que el programa pueda dibujar cada forma de manera correcta en un tamaño personalizado. Además, el programa debe:

- Evitar dibujar formas con dimensiones inferiores a 1
- Poder dibujar formas con cualquier dimensión igual a 1 (una forma de 1 x 1 debe imprimir un único carácter)

Si el problema parece complicado, intente dividirlo en pequeños retos, como:

- ¿Cómo se puede imprimir una única línea que sea un número variable de caracteres "#"?
- ¿Cómo se puede crear una cadena que empiece y acabe con un "#" pero tenga un número variable de espacios en medio?

Finalizar cada uno de estos pequeños retos es un logro. El problema radica en entender los bucles como un gran problema que se puede dividir en pequeñas tareas.

El conocimiento adquirido en la sección sobre bucles le resultará muy útil para completar este programa. Puede realizar cualquier tipo de sentencias de bucle que desee. También tendrá que tener en cuenta algunos conceptos de secciones anteriores.

Encontrará que los archivos `LoopShape.java` y `LoopShapeTest.java` son recursos útiles para empezar.