

Ejercicio 1 – Subida de Nivel

Crea un programa que pida al jugador su **nivel actual** (entero entre 1 y 50) y su **experiencia actual (XP)** (entero).

El programa debe mostrar cuánta experiencia le falta para subir al siguiente nivel, suponiendo que:

- La experiencia necesaria para subir de nivel se calcula como:
 $xpNecesaria = nivel * 1000$
- Si el jugador ya tiene suficiente XP, muestra un mensaje de “¡Has subido de nivel!” y aumenta su nivel en 1, dejando la experiencia sobrante.

Requisitos:

- Usa condicionales (if, else).
- Declara las variables con tipos adecuados (int, double).
- Muestra los resultados de forma clara.

Ejercicio 2 – Batalla contra enemigos

Vas a simular una batalla contra enemigos en un juego.

El jugador empieza con **100 puntos de vida**. En cada turno, se genera un enemigo aleatorio con un daño entre **5 y 25 puntos**.

El jugador puede elegir entre tres acciones:

- Atacar: quita entre 10 y 30 puntos de vida al enemigo.
- Defender: reduce el daño recibido a la mitad.
- Curarse: recupera entre 10 y 20 puntos de vida, sin superar 100

La batalla continúa hasta que la vida del jugador llegue a 0 o el jugador escriba “salir”.

Requisitos:

- Usa un **bucle while o do-while** para repetir turnos.
- Usa un **switch** para gestionar las acciones del jugador.
- Usa **funciones** para:
 - atacar(): devuelve el daño causado (aleatorio entre 10 y 30).
 - defender(): devuelve el daño recibido reducido a la mitad.

- curar(): devuelve los puntos de vida recuperados (aleatorio entre 10 y 20).
- Muestra en cada turno los puntos de vida actuales del jugador.
- Usa la clase Math para generar los valores aleatorios.

Pista: puedes usar Math.random() para generar los valores.

Para generar un número entero aleatorio entre 1 y numMax puedes reutilizar este código:

```
int valorDado = (int) Math.floor(Math.random() * numMax + 1);
```