

a) ¿Cuántas clases se están definiendo en este ejercicio?

sln/ Se estan definiendo 3 clases

b) ¿Para qué sirve la línea de código `if __ name = “ _main_ ”:?`

sln/ Se usa para controlar la ejecución de código de prueba o de ejemplo

c) ¿Qué sucede si retiro la línea de la pregunta anterior en nuestro código?,
¿Este sigue corriendo o hay error? Explique en ambos casos.

sln/ Si eliminas la línea, el comportamiento seguirá siendo el mismo.

*No habrá error, pero perderás la claridad y el control de que ese bloque es solo para pruebas o ejecución principal (Si se ejecuta directamente).

*El bloque de código que ahora está al final del archivo también se ejecutará al importar este archivo en otro script (importa como un módulo en otro programa).

d) ¿Cuántos objetos de la clase Apostador se están creando?

sln/ Se están creando 2 objetos.

e) ¿Cuáles objetos de la clase Apostador se están creando?

sln/ apostador1 y apostador2

f) ¿A quién está haciendo referencia la variable `self` de la línea 15 de la clase Apostador cuando se ejecuta el programa principal?

sln/ Hace referencia a la instancia de la clase Apostador que invoca el método `play`.

g) ¿Cuántos objetos de la clase Loteria se están creando?

- En la línea 4 del `main.py` cambiar el `apostador1.deposit(500)` por `apostador1.deposit(300)`

sln/ Se están creando 2 objetos.

h) ¿Qué imprimiría el código por parte del apostador1?

- En la línea 10 del `main.py` cambiar el `apostador2.deposit(500)` por `apostador2.deposit(400)`

sln/ 300
Necesitas poner mas dinero en tu wallet
300

i) ¿Qué imprimiría el código por parte del apostador2?

sln/ Hay dos posibles resultados:

*Si gana, se imprimirá algo como:

Has ganado X

*Si pierde, se imprimirá:

Has perdido lo que apostaste

0 (su saldo será reducido en 400)

j) ¿Cuáles atributos de la clase Lotería están haciendo referencia a objetos?

sln/ self.apostador y self.loteria

k) ¿Cuáles atributos de la clase Lotería están haciendo referencia a tipos primitivos?

sln/ self.value y self.probability

l) ¿Complete las siguientes líneas para que en la clase Loteria, se implemente el método de clase changeProbability?

```
- _____  
- def changeProbability(__, nprobability):  
-     _____ .probability = nprobability
```

```
sln/ @classmethod  
      def changeProbability(cls, nprobability):  
          cls.probability = nprobability
```

m) ¿Cómo sería la línea de código para llamar el método changeProbability?

sln/ Loteria.changeProbability(nprobability)

n) ¿Es correcto si en el método changeProbability que se creó, cambiar lo siguiente? Explique

Línea Original:

```
- cls.probability = nprobability
```

Línea Nueva:

```
- Loteria.probability = nprobability
```

sln/ La línea original usando es más general y flexible, Sin embargo, es totalmente válida en el caso de que no se necesite la flexibilidad de la herencia y quieras referenciar directamente la clase Loteria.

o) ¿Cuántos métodos tiene la clase Loteria después de agregarle el nuevo método?

sln/ tiene 5 métodos

p) ¿Si el apostador1 gana el apostador2 también? Explique por qué pasa en caso de ser sí o no

sln/ Si apostador1 gana, no implica que apostador2 gane también.
Ambos jugadores juegan de manera independiente, cada uno con su propia probabilidad y su propio resultado.

q) ¿Qué sucede si decido cambiar el atributo de clase probability a una constante?
¿Se considera correcto el uso del método changeProbability teniendo en cuenta este nuevo cambio?

sln/ Si probability es tratada como una constante, no se debe usar el método changeProbability, ya que su propósito es modificar el valor de probability

r) ¿Cuál es el tipo de retorno de los métodos gain() y commission() de la clase ComisionJuegoEspectaculos?

sln/ Devuelven un valor de tipo float

s) ¿A quién está haciendo referencia la variable self de la línea 18 de la clase Loteria cuando se ejecuta el programa principal? ¿Podría omitirse el uso de la variable self en este caso?

sln/ self hace referencia a la instancia actual de la clase Loteria.

No se puede omitir self en este caso si deseas acceder a los atributos de la clase, ya que self se usa para acceder a los miembros de la clase dentro de sus métodos

t) ¿En la línea 15 de la clase apostador vemos como la clase recibe dos parámetros (value, self) especificar cuál de estos pasa por valor y cuál por referencia y por qué?

sln/ value pasa por valor, ya que es un tipo primitivo
self pasa por referencia, porque es una instancia de la clase Apostador.