



RESPUESTAS

- a. Se están definiendo tres clases lotería, apostador, comisionjugosespectaculos
- b. Sirve para hacer que el código solo se ejecute cuando se hace directamente
- c. Al quitar esa línea del código main.py el programa seguirá corriendo con normalidad
- d. Se está creando 1 objeto de la clase apostador
- e. apostador1 y apostador2
- f. Está haciendo referencia a la variable apostador1
- g. Se están creando tres objetos de la clase lotería
- h. Cuando se realiza el cambio por parte del apostador1 imprime que "Necesitas poner más dinero en tu wallet"
- i. Después de reemplazar el valor por parte del apostador2 imprime que "Has perdido lo que apostaste"
- j. Dentro de la clase apostador el atributo que hace referencia a un objeto es self.apostador
- k. Los atributos de la clase lotería que hacen referencia a primitivos son probability y self.value
- l.
 - @classmethod
 - def changeProbability(cls, nprobability):
 - cls.probability = nprobability
- m. La línea de código donde vamos a llamar el método sería Loteria.changeprobability(0.5) aunque debemos importar la clase Lotería al principio del programa
- n. No, aunque no cambiaría el funcionamiento del programa lo correcto sería usar cls para tener un mejor orden
- o. Antes tenía un total de 4 más el nuevo método de clase son 5 en total
- p. Que el apostador1 gane no significa que el apostador2 también lo haga debido a que están programados para trabajar de forma independiente es decir que los resultados del uno no influyan en el otro, en el programa solo comparten el valor de probabilidad

- q. Si cambiamos el atributo de clase `probability` a una constante estaríamos programando de tal forma que el valor de `probability` se ha el mismo todo el tiempo para el código lo que haría inservible el método `changeprobability`
- r. El tipo de retorno de los métodos `gain()` y `commission()` viene de los valores de loteria y `percentage` por lo tanto son un numero tipo `int` o `float`
- s. Teniendo en cuenta el programa principal la variable `self` de la línea 18 de la clase `loteria` hace referencia a la instancia que se encuentra en la clase `loteria` en la cual se está llamando el método `playGame()` que es fundamental para correr el programa por lo cual es imposible omitir el uso de la variable `self`
- t. En la línea 15 de la clase `apostador` el parámetro `value` pasa por valor ya que esta trabajando con el método `playGame()` que contiene el valor del apostador y el parámetro `self` pasa por referencia ya que esta referenciando a la variable de instancia de `apostador`