

## Taller 2 Python

Pablo Bernal García

1025888826

Grupo 2

### Taller 2 Python – Respuestas

- a) Se están definiendo cuatro clases: Apostador, ComisionJuegoEspectaculos, Loteria y la clase implícita main.
- b) Esta línea se usa para que el bloque de código correspondiente solo e ejecute cuando el archivo se ejecuta directamente y no si se importa.
- c) Si se elimina está línea, entonces el código de main se va a ejecutar cada que este sea importado por otro módulo.
- d) En la clase main, se crean dos objetos de clase Apostador.
- e) Se crean apostador1 y apostador2.
- f) Hace referencia al propio objeto Apostador que está llamando al método play().
- g) Antes del cambio se crea una instancia de Loteria al poder invocar el método play() de la clase Apostador. Después del cambio no se crea

ninguna y se imprime "Necesitas poner mas dinero en tu wallet" porque no se puede invocar el método play() al no tener suficiente saldo.

h) Se imprime su saldo y si gana se imprime que ha ganado y luego se imprime su nuevo saldo en la wallet después de recibir la ganancia y la comisión. Si pierde, se imprime que perdió y se imprime su nuevo saldo que sería de

$$500 - 400 = 100$$

i) Se imprime su saldo y si gana se imprime que ha ganado y luego se imprime su nuevo saldo en la wallet después de recibir la ganancia y la comisión. Si pierde, se imprime que ha perdido y se da su nuevo saldo que sería igual a 0 y no podría seguir jugando.

j) El atributo apostador de la clase de Loteria es el único a hacer referencia a otro objeto, siendo este de la clase Apostador.

k) Los atributos de tipo primitivo de la clase Loteria son value: Integer y probability: Float.

l) @classmethod

```
def changeProbability(cls, nprobability):
```

```
    cls.probability = nprobability
```

m) Loteria.changeProbability(nprobability)

- n) Ambas líneas funcionan igual porque probability es un atributo de clase y no de instancia y, en Python los atributos de clase pueden ser accedidos y modificados directamente a través del nombre de la clase.
- o) Después de agregar el nuevo método, la clase lotería tendría cuatro métodos: `__init__`, `payMoney`, `recieveMoney`, `playGame` y `changeProbability`.
- p) No, cada instancia de Apostador juega individualmente y el resultado depende de la ejecución individual del método `playGame()` en la instancia correspondiente de Loteria.
- q) No sería apropiado, pues por convención las constantes deben mantenerse fijas.
- r) Los dos métodos retornan valores de tipo float debido a la multiplicación con un porcentaje decimal, lo que lo convierte automáticamente en float.
- s) Hace referencia al objeto actual de la clase Loteria que está ejecutando el método `playGame()`. No es posible omitirlo ya que el constructor de `ComisionJuegosEspectaculos` no recibiría el objeto de Loteria que necesita para su funcionamiento.

t) Value pasa por valor, pues es un tipo de dato primitivo y en Python estos se pasan por valor. Self se pasa por referencia, pues hace referencia a la instancia actual de Apostador.