

1

PROGRAMACIÓN ORIENTADA A OBJETOS - UNALMED 2024 -2

Preguntas de análisis

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

3

b) ¿Para qué sirve la línea de código `if __name__ == "__main__":`?

Para confirmar si el programa actual se está corriendo como main

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

Sigue corriendo, el intérprete correrá el programa línea a línea, lo que nos causaría conflicto si existiera otro código libre, pero como nuestro main es el único código ejecutable asumirá como main

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

2

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

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?

al objeto de clase Apostador

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

2

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

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

Le indica que necesita más dinero en su wallet y retorna el dinero inicial (300)

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

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

apostarí todo, por lo que sería un todo o nada para el apostador2

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

apostador

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

value

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

- probability = 0.5

- def changeProbability(cls, nprobability):

- cls.probability=nprobability

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

Loteria.changeProbability(self, value)

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

El código seguiría funcionando ya que es un método de clase, pero es recomendable usar el cls para mantener el código genérico, pues cls llama a la clase misma

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

4

p) ¿Si el apostador1 gana el apostador2 también? Explique por qué pasa en caso de ser sí o

no

no, son casos totalmente independientes

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?

podría usarse, ya que nprobability es un primitivo siempre

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

ComisionJuegoEspectaculos?

Floats

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?

al objeto de la clase Loteria

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?

value entra como valor al ser un entero inmutable y self como referencia, pues se refiere a una instancia de la clase Apostador