

# Curso de introducción a la programación (Glovo)

Puntos totales 22/30

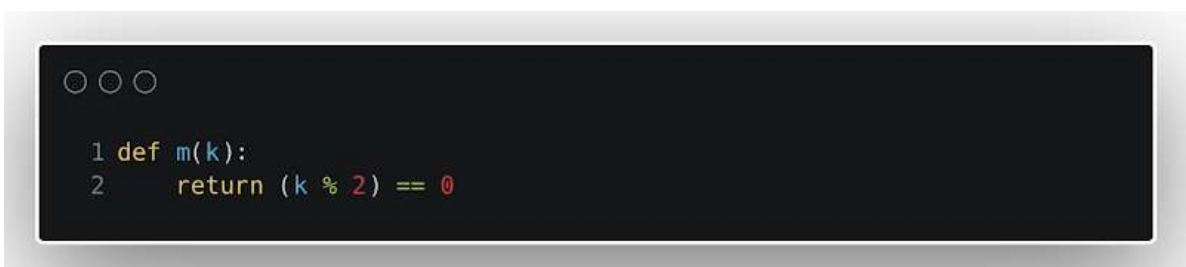
Este test no tiene tiempo límite de realización y puedes seguir con él si tuvieras que abandonarlo. Lo que sí tiene es una fecha límite de entrega 18/09/2022 a las 23:55 hora peninsular española

Correo \*

dianamtlopez@hotmail.com

Pregunta nº 1

1 de 1 puntos



```
1 def m(k):  
2     return (k % 2) == 0
```

✓ Explica lo que hace la función de la imagen \*

1/1

- ☐ Divide su parámetro por 2
- ☐ Multiplica k por 2
- ☐ Nada, esa función no compila



☒ Comprueba si k es un número par



Pregunta nº 2

1 de 1 puntos

○ ○ ○

```
1 def f(ht):
2     b = 0
3     for elt in ht:
4         b = elt + b
5     return b
```

✓ Explica lo que hace la función de la imagen \*

1/1

- ☐ La función f recibe un número y devuelve la suma de dicho número y de la variable b
- ☒ La función f recibe una lista y devuelve la suma de sus elementos ✓
- ☐ La función f recibe una lista y devuelve su longitud
- ☐ La función f recibe una lista y devuelve otra lista con la suma de cada elemento en ht más el valor de b

Pregunta nº 3

0 de 1 puntos

○ ○ ○

```
1 def s(m, n):
2     r = None
3     if m <= n:
4         r = m
5     else:
6         r = n
7     return r
```



✗ Explica lo que hace la función de la imagen. Si llamo a s pasándole 2 números... \*0/1

- ☐ Me devolverá el menor de ellos.
- ☒ Me devolverá el mayor de ellos. ✗
- ☐ Devolverá siempre None.
- ☐ Dará un error si ambos números son iguales

Respuesta correcta

- ☒ Me devolverá el menor de ellos.

Pregunta nº 4

1 de 1 puntos

○ ○ ○

```
1 def interes_anual(cantidad, interes):
2     return cantidad * (1 + interes / 100)
3
4 def retorno_inversion(cantidad, interes, anyos):
5     r = []
6     for anyo in range(anyos):
7         cantidad = interes_anual(cantidad, interes)
8         r.append(round(cantidad, 2))
9
10    return r
```



✓ Según el código mostrado ¿qué devolverá la siguiente instrucción  
retorno\_inversion(10, 4, 3)?

\*1/1

- ☐ [10, 4, 3]
- ☒ [10.4, 10.82, 11.25]
- ☐ 11.25
- ☐ Ninguna es correcta



Pregunta nº 5

0 de 1 puntos

○ ○ ○

```
1 def applyToEachString(stringFunction, *words):
2     myList = []
3     for word in words:
4         # ¿Qué instrucción debe ir aquí?
5     return tuple(myList)
6
7 def onlyVowels(word):
8     vowels = 'aeiouáéíóúü'
9     resp = ""
10    for letter in word:
11        if letter in vowels or letter in vowels:
12            resp = resp + letter
13
14    return resp
```



✗ Por cual de las siguientes instrucciones debes sustituir el comentario en \*0/1  
applyToEachString para que el resultado de  
applyToEachString(onlyVowels, "Periódico", "Navío", "Computadora") sea  
("eióio", "aío", "ouaoa")

- ☐ myList = stringFunction(word)
- ☒ myList.append(onlyVowels(word))
- ☐ myList.append(stringFunction(word))
- ☐ myList.append(stringFunction, word)

✗

Respuesta correcta

- ☒ myList.append(stringFunction(word))

Pregunta nº 6

0 de 1 puntos

```
1 def applyToEachString(stringFunction, *words):
2     myList = []
3     for word in words:
4         # ¿Qué instrucción debe ir aquí?
5     return tuple(myList)
```



✗ Con la función `applyToEachString` tal y como quedaría en la pregunta anterior, ¿cómo la llamarías ahora para que su resultado fuese ("PERIÓDICO", "NAVÍO", "COMPUTADORA").

\*0/1

- ☐ `applyToEachString(upper, "Periódico", "navío", "computadora")`
- ☒ `applyToEachString(str.upper, ("Periódico", "navío", "computadora"))` ✗
- ☐ `applyToEachString(str.upper(), "Periodico", "navío", "computadora")`
- ☐ `applyToEachString(str.upper, "Periódico", "navío", "computadora")`

Respuesta correcta

- ☒ `applyToEachString(str.upper, "Periódico", "navío", "computadora")`

Pregunta nº 7

1 de 1 puntos

```
1 def what_i_do(p1, p2):
2     if p2 < len(p1):
3         return p1
4
5     gap = (p2 - len(p1)) // 2
6
7     return " " * gap + p1
```



✓ Explica lo que hace la función de la imagen \*

1/1

- ☐ ordenar dos cadenas.
- ☐ comparar dos cadenas y devolver la mayor con tantos espacios delante como la longitud de la menor.
- ☒ centrar un texto en un ancho determinado, p1 es la palabra y p2 el ancho donde centrarla ✓
- ☐ separar dos palabras por un número determinado de espacios

Pregunta nº 8

0 de 1 puntos





```
1 def second_scored(*lis):
2     maximum = max(lis)
3     preMax = min(lis)
4     for nombre in lis:
5         if (nombre < maximum and nombre > preMax):
6             preMax = nombre
7     return preMax
8
9 """
10 -EL COMENTARIO-
11 """
12
13 def another_second_scored(*lis):
14     arr = list(set(lis))
15     arr.sort()
16     return arr[-2]
17
18 def other_second_scored(*lis):
19     for n in lis:
20         i = 2
21         is_second = True
22         while i <= n // 2:
23             if n % i == 0:
24                 is_second = False
25             i += 1
26
27         if is_second:
28             return n
29
30 def the_second_scored(*lis):
31     first = max(lis)
32     lis = list(lis)
33     lis.remove(first)
34     while first == max(lis):
35         lis.remove(first)
36     return max(lis)
37
38
```





✗ De las tres funciones tras el comentario ¿cuál de ellas hace lo mismo que `*0/1 second_scored(*lis)`?

- ☒ the\_second\_scored ✗
- ☐ the\_second\_scored y other\_second\_scored
- ☐ another\_second\_scored
- ☐ the\_second\_scored y another\_second\_scored

Respuesta correcta

- ☒ the\_second\_scored y another\_second\_scored

Pregunta nº 9

1 de 1 puntos

```
1 def n(a):
2     return (a % 2) == 0
3
4 def p(k):
5     return not n(k)
```

✓ Contesta a las preguntas sobre los parámetros y valores de retorno de estas funciones. \*1/1

- ☒ La función p devuelve un booleano ✓
- ☐ La función p devuelve un número entero

- ☐ La función p devuelve un número entero
- ☐ La función p devuelve una lista
- ☐ La función p devuelve un error

Pregunta nº 10

1 de 1 puntos

```
1 def g(xxs):
2     k = 1
3     for elm in xxs:
4         k = k * elm
5     return k
```

✓ Contesta a las preguntas sobre los parámetros y valores de retorno de estas funciones. \*1/1

- ☐ La función g recibe un número y devuelve otro número.
- ☐ La función g recibe una lista de números y devuelve otra lista.
- ☒ La función g recibe una lista de números y devuelve un número. ✓
- ☐ La función g no compila.

Pregunta nº 11

1 de 1 puntos

```
1 for x in ['Darth Vader', 'Luke Skywalker', 'Obi Wan', 'Leia Organa', 'R2-D2', 'C-3P0']:
2     pass
```

✓ Qué afirmación es la correcta. \*

1/1

- ☒ El bucle for itera por cada cadena. ✓
- ☐ El bucle for itera por cada letra de cada cadena.
- ☐ El bucle for no es válido y dará un error.

Pregunta nº 12

1 de 1 puntos

```
1 # bucle 1
2 for x in range(0,6):
3     pass
4
5 # bucle 2
6 index = 0
7 while index <= 6:
8     pass
9     index = index + 1
```

✓ Qué afirmación es la correcta. \*

1/1

- ☐ El bucle 1 y el bucle 2 son equivalentes y se ejecutan el mismo número de veces.
- ☐ El bucle 1 y el 2 NO son equivalentes. El bucle for se ejecuta más veces que el while.
- ☒ El bucle 1 y el 2 NO son equivalentes. El bucle while se ejecuta más veces que el for ✓



Pregunta nº 13

1 de 1 puntos

```
1 for x in range(0,9):  
2     pass
```

✓ Qué afirmación es la correcta. \*

1/1

- ☐ El bucle for se ejecuta 10 veces, de 0 a 9.
- ☒ El bucle for se ejecuta 9 veces, de 0 a 8.
- ☐ El bucle for se ejecuta 10 veces, de 1 a 10.



Pregunta nº 14

1 de 1 puntos

✓ Si recibes una lista de números y tienes que producir una nueva lista con cada uno de los números originales multiplicado por 2, lo más sencillo será usar un: \*1/1

- ☒ Un bucle for.
- ☐ Un bucle while.



Pregunta nº 15

1 de 1 puntos



✓ Tienes que recibir números que el usuario escribe en el terminal, multiplicarlo por dos e imprimirlo en pantalla. Debes de hacer esto, hasta que el usuario escriba un 0. Para ello, lo más sencillos será usar:

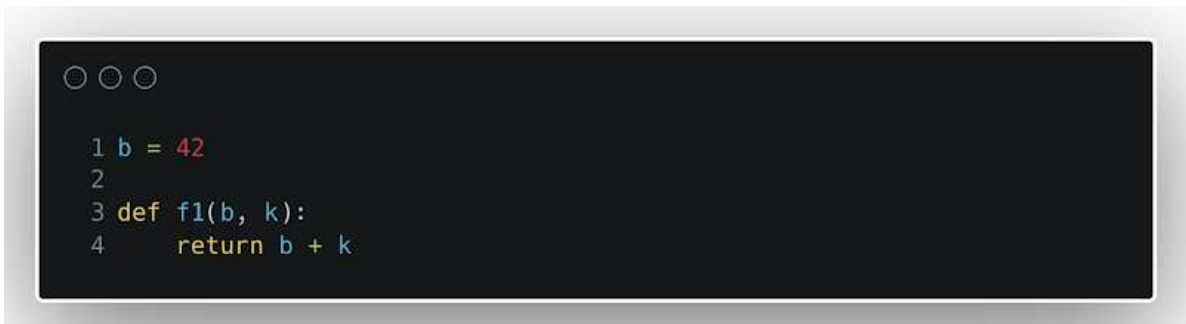
\*1/1

- ☐ Un bucle for.
- ☒ Un bucle while.



Pregunta nº 16

1 de 1 puntos



```
1 b = 42
2
3 def f1(b, k):
4     return b + k
```

✓ Cuando llamo a f1(0,8), me devuelve: \*

1/1

- ☐ 50
- ☒ 8



Pregunta nº 17

1 de 1 puntos



○ ○ ○

```
1 q = 51
2 def np(a):
3     q = 42 # ¿qué ocurre aquí?
4     return q * a
```

✓ Cuando dentro de la función np se llega a la línea marcada por el comentario, ¿qué ocurre?

\*1/1

- ☐ Se modifica el valor anterior de q, que de valer 51 pasa a valer 42.
- ☒ Se crea una nueva variable, llamada q, sólo visible dentro de la función np, con un valor de 42. ✓

Pregunta nº 18

0 de 1 puntos

○ ○ ○

```
1 z = -9
2 def ik(z):
3     return z
```



✗ Cuando llamo a ik(42), me devolverá: \*

0/1

☐ 42

☒ -9

☐ 33

✗

Respuesta correcta

☒ 42

Pregunta nº 19

0 de 1 puntos

○ ○ ○

```
1 ['Three',  
2 'Rings',  
3 'for',  
4 'the',  
5 'Elven',  
6 'Lords',  
7 'under',  
8 'the',  
9 'Sky',  
10 'Seven',  
11 'for',  
12 'the',  
13 'Dwarf',  
14 'Lords',  
15 'in',  
16 'their',  
17 'halls',  
18 'of',  
19 'Stone',  
20 'Nine',  
21 'for',  
22 'Mortal',  
23 'Men',
```



```
24 'doomed',  
25 'to',  
26 'die']
```

✗ Tienes que analizar un fichero de texto con el contenido del "Señor de los Anillos", para descubrir cuales son las palabras más usadas en dicha obra. El fichero es texto puro, sin ningún formato y ya lo tienes en una lista, como la de la imagen. Tendrás que iterar por esa lista para ir calculando la frecuencia de cada palabra. Para llevar esa cuenta, ¿qué estructura de datos usarías? \*0/1

- ☒ Otra lista. ✗
- ☐ Un diccionario (dict).
- ☐ Un conjunto (set).
- ☐ Una lista de listas.

Respuesta correcta

- ☒ Un diccionario (dict).

Pregunta nº 20

0 de 1 puntos

✗ Tienes que almacenar en tu programa todos los valores diarios de las acciones de Tesla a lo largo de un año, para poder luego calcular promedios y hacer proyecciones. ¿Qué estructura de datos usarías para almacenar dichos precios? \*0/1

- ☒ Un diccionario (dict). ✗
- ☐ Una lista.
- ☐ Una tupla.
- ☐ Un conjunto (set).





Respuesta correcta

- ☒ Una lista.

Pregunta nº 21

1 de 1 puntos

○ ○ ○

```
1 fruits = {'banana', 'apple', 'orange', 'cherry'}
```

✓ fruit es: \*

1/1

- ☐ Una lista.
- ☐ Un diccionario de cadenas.
- ☒ Un set de cadenas.



Pregunta nº 22

1 de 1 puntos

○ ○ ○

```
1 fruits = {'banana', 'apple', 'orange', 'cherry'}  
2 fruits.add('orange') # ¿qué hará esta línea?
```

✓ ¿Que hará la linea comentada? \*

1/1

- ☐ Añadirá otra cadena 'orange' al principio de fruits.
- ☐ Añadirá otra cadena 'orange' al final de fruits.



- ☒ No hará nada, puesto que la cadena 'orange' ya está presente en fruits.



Pregunta nº 23

0 de 1 puntos

✗ Imagina ahora que quieres obtener la lista de todos los caracteres que forman una palabra. ¿Qué estructura de datos es mejor para ello? \*0/1

- ☐ Un diccionario (dict).
- ☒ Una lista.
- ☐ Una tupla.
- ☐ Un conjunto (set).



Respuesta correcta

- ☒ Un conjunto (set).

Pregunta nº 24

1 de 1 puntos

```
1 def adder(n):
2     return lambda x : x + n
```

✓ Elige la afirmación correcta \*

1/1

- ☐ La función adder recibe un número (n) y devuelve la suma de x y n.
- ☒ La función adder recibe un número y devuelve una función que a su vez tomará otro número (x) y lo suma a n. ✓
- ☐ La función adder recibe una función y devuelve un número.



Pregunta nº 25

1 de 1 puntos

○ ○ ○

```
1 def adder(n):  
2     return lambda x : x + n  
3  
4 f = adder(2)
```

✓ Elige la afirmación correcta. \*

1/1

- ☐ f(8) devolverá un error.
- ☐ f(3, 2) devolverá 5.
- ☒ f(2) devolverá 4.



Pregunta nº 26

1 de 1 puntos

○ ○ ○

```
1 def g(f):  
2     def k(*args):  
3         return 2 * f(*args)  
4     return k  
5
```



✓ Elige la afirmación correcta \*

1/1

- ☐ g es una función que recibe como parámetro otra función y devuelve un número.
- ☐ g es una función, que recibe como parámetro un número y devuelve otra función.
- ☒ g es una función, que recibe como parámetro otra función y devuelve otra función. ✓

Pregunta nº 27

1 de 1 puntos

○ ○ ○

```
1 # Jerarquía de clases para representar personajes de Star Wars
2 class StarWarsCharacter:
3     def __init__(self, name):
4         self.name = name
5
6 class Jedi (StarWarsCharacter):
7     def __init__(self, name, midichlorians):
8         super().__init__(name)
9         self.midichlorians = midichlorians
10
11
```

✓ Elige la afirmación correcta. \*

1/1

- ☒ Jedi es una subclase de StarWarsCharacter. ✓
- ☐ StarWarsCharacter es una subclase de Jedi.
- ☐ Jedi es una instancia de la clase StarWarsCharacter.



```
class StarWarsCharacter:
    def __init__(self, name):
        self.name = name

class Jedi (StarWarsCharacter):
    def __init__(self, name, midichlorians):
        super().__init__(name)
        self.midichlorians = midichlorians

chewie = StarWarsCharacter('Chewbacca')
m = chewie.midichlorians
```

✓ Elige la afirmación correcta. \*

1/1

- ☐ m contendrá el valor 100.
- ☐ m contendrá None.
- ☒ Ese código provocará un error.



```
class StarWarsCharacter:
    def __init__(self, name):
        self.name = name

class Jedi (StarWarsCharacter):
    def __init__(self, name, midichlorians):
        super().__init__(name)
        self.midichlorians = midichlorians

anakin = Jedi('Anakin', 100000)
```

✓ Elige la afirmación correcta. \*

1/1

- ☐ Si accedemos a la propiedad name de anakin, nos dará un error, puesto que name es una propiedad de StarWarsCharacter y no Jedi.
- ☐ La clase anakin tiene una propiedad name, puesto que hereda de la clase StarWarsCharacter.
- ☒ La instancia anakin tiene una propiedad name, definida en la clase StarWarsCharacter, de la cual hereda. ✓

○ ○ ○

```
1 characters = [Jedi('Anakin', 10000), Jedi('Yoda', 1000000), StarWarsCharacter('Chewie')]
2 sum = 0
3 for elt in characters:
4     sum = sum + elt.midichlorians
```

✓ Elige la afirmación correcta. \*

1/1

- ☐ Ese código va sumando los midiclorianos de cada uno de los personajes y en sum habrá un número.
- ☒ Ese código va a causar un error cuando intente acceder a la propiedad midichlorians del último objeto de la lista. ✓

Este formulario se creó en KeepCoding.

Google Formularios

