

✓ ¡Felicitaciones! ¡Aprobaste!

Calificación recibida 100 % Para Aprobar 75 % o más

[Ir al siguiente elemento](#)

1. Completa el espacio en blanco: En el siguiente código, los números enteros 5 y 12 son _____:

1 / 1 punto

```
for i in range(5, 12):
```

```
print(i)
```

- ☒ argumentos
- ☐ funciones
- ☐ parámetros
- ☐ sentencias return

✓ Correcto

Los números enteros 5 y 12 son argumentos en el siguiente código:

```
for i in range(5, 12):
```

```
print(i)
```

Los argumentos son los datos que se introducen a una función cuando se la llama. En este caso, 5 y 12 se ingresan a la función `range()` cuando se la llama.

2. ¿Cuál es la forma correcta de definir la función `addition()` si requiere los dos parámetros `num1` y `num2`?

1 / 1 punto

- ☐ `def addition(num1 and num2):`
- ☒ `def addition(num1, num2):`
- ☐ `def addition(num1 num2):`
- ☐ `def addition(num1) (num2):`

✓ Correcto

La forma correcta de definir la función `addition()` si requiere los dos parámetros `num1` y `num2` es `def addition(num1, num2):`. Si una función necesita varios parámetros, al definir la función, estos deben colocarse entre paréntesis y separarse con comas.

3. ¿Cuál de las siguientes líneas de código tiene la sintaxis correcta para imprimir el tipo de dato de la cadena `"elarson"`?

1 / 1 punto

- ☐ `type(print("elarson"))`
- ☐ `print(type, "elarson")`
- ☐ `print("elarson", type)`
- ☒ `print(type("elarson"))`

✓ Correcto

El código `print(type("elarson"))` tiene la sintaxis correcta para imprimir el tipo de dato de la cadena `"elarson"`. Primero se procesa la función interna y luego su valor devuelto se pasa a la función externa. El argumento `"elarson"` se pasa primero a la función `type()`. Esta función devuelve el tipo de dato y después se pasa a la función `print()`.

4. ¿Qué definición de función incluye la sintaxis correcta para devolver el valor de la variable `result` de la función `doubles()`?

1 / 1 punto

- ☐ `def doubles(num):`
`result = num * 2`
`return "result"`
- ☐ `def doubles(num):`
`result = num * 2`

```
result return
```

☒ `def doubles(num):`

```
result = num * 2
```

```
return result
```

☐ `def doubles(num):`

```
result = num * 2
```

```
return = result
```

☒ **Correcto**

El siguiente bloque de código demuestra la sintaxis correcta para devolver el valor de la variable `result` desde la función `doubles()`:

```
def doubles(num):
```

```
result = num * 2
```

```
return result
```

La palabra clave `return` se utiliza para devolver información desde una función. Se coloca antes de la información que se desea devolver. En este caso, se trata de la variable `result`.