

# Definir una función en Python

Nombre: \_\_\_\_\_

Fecha: \_\_\_\_\_

Puntaje: \_\_\_\_\_

---

**1.**

¿Cuál es la palabra clave utilizada para definir una función en Python?

1. func
2. define
3. def
4. lambda

**Respuesta correcta:**

**C.**

def

**2.**

¿Cuál de las siguientes es la sintaxis correcta para definir una función llamada 'saludar' que imprime 'Hola'?

1. def saludar(): print('Hola')
2. function saludar(): print('Hola')
3. def saludar: print('Hola')
4. def saludar() print('Hola')

**Respuesta correcta:**

**A.**

def saludar(): print('Hola')

**3.**

Después de la línea de cabecera de una función (por ejemplo, `def saludar():`), ¿qué debe venir a continuación?

1. Una línea en blanco
2. Un bloque de código indentado
3. Un comentario

**Respuesta correcta:**

**B.**

Un bloque de código indentado

**4.**

Para llamar a la función 'saludar', debes escribir: `__` (incluyendo paréntesis)

Respuesta: \_\_\_\_\_

**Respuesta correcta:**

`saludar()`

**5.**

¿Qué imprime el siguiente código?

```
def saludar():  
    print('Hola')  
saludar()
```

1. Nada
2. Hola
3. Error de sintaxis
4. `saludar`

**Respuesta correcta:**

**B.**

Hola

**6.**

Si defines una función pero nunca la llamas, ¿qué ocurre cuando ejecutas el programa?

1. La función se ejecuta automáticamente
2. No ocurre nada hasta que la llames
3. Ocurre un error
4. El programa imprime la definición de la función

**Respuesta correcta:**

**B.**

No ocurre nada hasta que la llames

**7.**

¿Qué valor devuelve por defecto una función que no contiene una sentencia return?

Respuesta: \_\_\_\_\_

**Respuesta correcta:**

None

**8.**

¿Cuál es la forma correcta de documentar una función usando un docstring?

1. Esta función imprime un saludo
2. ""Esta función imprime un saludo"" (colocado como primera línea dentro de la función)
3. 'Esta función imprime un saludo'
4. // Esta función imprime un saludo

**Respuesta correcta:**

**B.**

""Esta función imprime un saludo"" (colocado como primera línea dentro de la función)

**9.**

¿Cuál de los siguientes nombres de función sigue la convención de estilo snake\_case?

1. saludoBienvenida
2. saludo-bienvenida
3. saludo\_bienvenida
4. SaludoBienvenida

**Respuesta correcta:**

**C.**

saludo\_bienvenida

**10.**

Si olvidas los dos puntos al final de la cabecera de una función (ej. def saludar()), ¿qué tipo de error se produce?

Respuesta: \_\_\_\_\_

**Respuesta correcta:**

SyntaxError

**11.**

¿Qué sucede si llamas a una función como 'saludar' sin los paréntesis?

1. La función se ejecuta
2. Se obtiene una referencia al objeto función
3. Ocurre un error
4. No pasa nada

**Respuesta correcta:**

**B.**

Se obtiene una referencia al objeto función

**12.**

¿Cuál de las siguientes funciones mostrará 'Hola' en pantalla al ser llamada?

1. `def f(): return 'Hola'`
2. `def f(): print('Hola')`
3. `def f(): 'Hola'`
4. `def f(): print('Hola') y return`

**Respuesta correcta:**

**B.**

```
def f(): print('Hola')
```

**13.**

¿Cuál de las siguientes definiciones de función es válida si se desea incluir dos instrucciones?

1. `def f(): print(1); print(2)`
2. `def f(): print(1) print(2)`
3. `def f(): print(1), print(2)`
4. `def f(): print(1) then print(2)`

**Respuesta correcta:**

**A.**

```
def f(): print(1); print(2)
```

**14.**

La indentación estándar recomendada en Python es de \_\_\_\_ espacios.

Respuesta: \_\_\_\_\_

**Respuesta correcta:**

4

**15.**

¿Cuál de las siguientes define correctamente una función que imprime 'Buenos días'?

1. `def buenos_dias(): print('Buenos días')`
2. `def buenos_dias: print('Buenos días')`
3. `def buenos_dias(): print(Buenos días)`
4. `def buenos_dias() print('Buenos días')`

**Respuesta correcta:**

**A.**

```
def buenos_dias(): print('Buenos días')
```

**16.**

Para definir una función, el bloque de código dentro de la función debe estar \_\_ (una palabra).

Respuesta: \_\_\_\_\_

**Respuesta correcta:**

indentado

**17.**

Si llamas a una función exactamente una vez, ¿cuántas veces se ejecuta el cuerpo de la función?

1. 0
2. 1
3. 2

**Respuesta correcta:**

**B.**

1

**18.**

En Python, los dos puntos al final de la cabecera de una función indican:

1. El final de la función
2. El inicio de un bloque indentado
3. Un comentario
4. Un separador opcional

**Respuesta correcta:**

**B.**

El inicio de un bloque indentado

**19.**

¿Puede una función contener más de una instrucción print?

1. Sí
2. No
3. Solo si usa return

**Respuesta correcta:**

**A.**

Sí

**20.**

¿Cuál de los siguientes NO es un nombre de función válido en Python?

1. \_saludar
2. saludar1
3. 1saludar
4. saludar\_nuevo

**Respuesta correcta:**

**C.**

1saludar

## Respuestas

1. **C.**

def

2. **A.**

def saludar(): print('Hola')

3. **B.**

Un bloque de código indentado

4. saludar()

5. **B.**

Hola

6. **B.**

No ocurre nada hasta que la llames

7. None

8. **B.**

"""Esta función imprime un saludo""" (colocado como primera línea dentro de la función)

9. **C.**

saludo\_bienvenida

10. SyntaxError

11. **B.**

Se obtiene una referencia al objeto función

12. **B.**

def f(): print('Hola')

13. **A.**

def f(): print(1); print(2)

14. 4

15. **A.**

```
def buenos_dias(): print('Buenos días')
```

16. indentado

17. **B.**

1

18. **B.**

El inicio de un bloque indentado

19. **A.**

Sí

20. **C.**

1saludar