

✓ ¡Felicitaciones! ¡Aprobaste!

Calificación recibida 100 % Para Aprobar 75 % o más

Ir al siguiente elemento

1. ¿Qué tipos de errores se pueden encontrar al depurar un código? Selecciona tres respuestas.

1 / 1 punto

☐ Iterativas

☒ Excepciones

✓ Correcto

Los errores de sintaxis, los errores lógicos y las excepciones son todos los tipos de errores que puedes encontrar al depurar un código. Los errores de sintaxis implican un uso no válido del lenguaje Python. Los errores lógicos pueden no causar mensajes de error, pero sí producen resultados no deseados. Las excepciones ocurren cuando el programa no sabe cómo ejecutar el código a pesar de que es sintácticamente correcto.

☒ Errores lógicos

✓ Correcto

Los errores de sintaxis, los errores lógicos y las excepciones son todos los tipos de errores que puedes encontrar al depurar un código. Los errores de sintaxis implican un uso no válido del lenguaje Python. Los errores lógicos pueden no causar mensajes de error, pero sí producen resultados no deseados. Las excepciones ocurren cuando el programa no sabe cómo ejecutar el código a pesar de que es sintácticamente correcto.

☒ Errores de sintaxis

✓ Correcto

Los errores de sintaxis, los errores lógicos y las excepciones son todos los tipos de errores que puedes encontrar al depurar un código. Los errores de sintaxis implican un uso no válido del lenguaje Python. Los errores lógicos pueden no causar mensajes de error, pero sí producen resultados no deseados. Las excepciones ocurren cuando el programa no sabe cómo ejecutar el código a pesar de que es sintácticamente correcto.

2. El propósito de este código es indicar si es necesario actualizar un sistema operativo en particular. Sin embargo, contiene un error de sintaxis. Ejecuta este código, analiza la salida y, luego, depúralo. (Si quieres deshacer los cambios en el código, puedes hacer clic en el botón Reset o Restablecer).

1 / 1 punto

```
1 operating_system = "OS 2"
2 if operating_system == "OS 1":
3     print("updates needed")
4 elif operating_system == "OS 2":
5     print("no updates needed")
```

Ejecutar

Restablecer

no updates needed

Según lo que descubriste, ¿cómo puedes corregir el error?

☐ Eliminar todos los dos puntos (:).

☐ Sangrar la sentencia `elsif`.

☒ Cambiando la palabra clave `elsif` a `elif`.

☐ Usar un solo signo de igual (=) y no dos signos de igual (==).

✓ Correcto

Al ejecutar este código, el mensaje de error puede ayudarte a identificar el error de sintaxis y el número de línea donde se produce. Cambiar la palabra clave `elsif` a `elif` corregirá el error. Los errores de sintaxis implican el uso no válido del lenguaje Python, por ejemplo, una falta de ortografía en una palabra clave. La ortografía correcta para la palabra clave necesaria antes de la condición `operating_system == "OS 2"` es `elif`.

3. Escribiste un código que asigna tickets de incidentes de seguridad al equipo de ciberseguridad apropiado en función de su nivel de prioridad. Si el nivel de prioridad es 1, debe reenviarse al equipo A. Si el nivel de prioridad es 2, debe reenviarse al equipo B. Al probar el código, detectas que un incidente con nivel de prioridad 2 se reenvía al equipo A en lugar de al equipo B. ¿Qué tipo de error es este?

1 / 1 punto

☐ Error de sintaxis

☐ Excepción

☒ Error lógico

☐ Error de nombre

✓ **Correcto**

Este es un error lógico. Los errores lógicos se producen cuando la lógica utilizada en el código produce resultados no deseados. En esta situación, debido a que el ticket de incidente de seguridad se reenvía al equipo incorrecto, hay un resultado no deseado.

4. Escribiste un código que utiliza un algoritmo de búsqueda para encontrar la dirección IP de un empleado. Al probar el código, un mensaje de error indica que se está accediendo a un índice desconocido. ¿Qué tipo de error es este?

1 / 1 punto

- ☒ Excepción
- ☐ Error lógico
- ☐ Error de sintaxis
- ☐ Iterativa

✓ **Correcto**

Esta es una excepción. Las excepciones ocurren cuando Python no sabe cómo ejecutar el código a pesar de que es sintácticamente correcto. Esto sucede si le pides a Python que acceda a un índice que no existe.