

El manejo de excepciones permite capturar y gestionar errores durante la ejecución del programa sin que este se interrumpa. Utilizando bloques `try`, `except`, `else` y `finally`, Python permite anticiparse a posibles errores, garantizando una experiencia de usuario más controlada y robusta.

El bloque `try` intenta ejecutar el código y, si ocurre un error, el bloque `except` lo captura y permite manejarlo. Esta estructura permite que los errores se resuelvan sin que el programa finalice abruptamente, lo cual es fundamental para aplicaciones que deben seguir ejecutándose en condiciones adversas.

Las excepciones personalizadas se crean al definir nuevas clases de excepción, permitiendo manejar errores específicos de un contexto particular. Esto mejora la claridad y el control del flujo de errores, al hacer que el código sea más expresivo y enfocado en los problemas relevantes de la aplicación.

El bloque `else` se ejecuta si no ocurre ninguna excepción, mientras que el bloque `finally` se ejecuta siempre, independientemente de si hubo un error o no. `finally` es útil para realizar tareas de limpieza, como cerrar archivos o liberar recursos, asegurando que el sistema regrese a un estado seguro tras la ejecución.

La propagación de excepciones permite que los errores se transmitan a niveles superiores del código, donde pueden ser manejados de manera más adecuada. Al controlar cómo se propagan las excepciones, es posible decidir si una excepción debe ser tratada en el lugar donde ocurre o en un nivel más global.