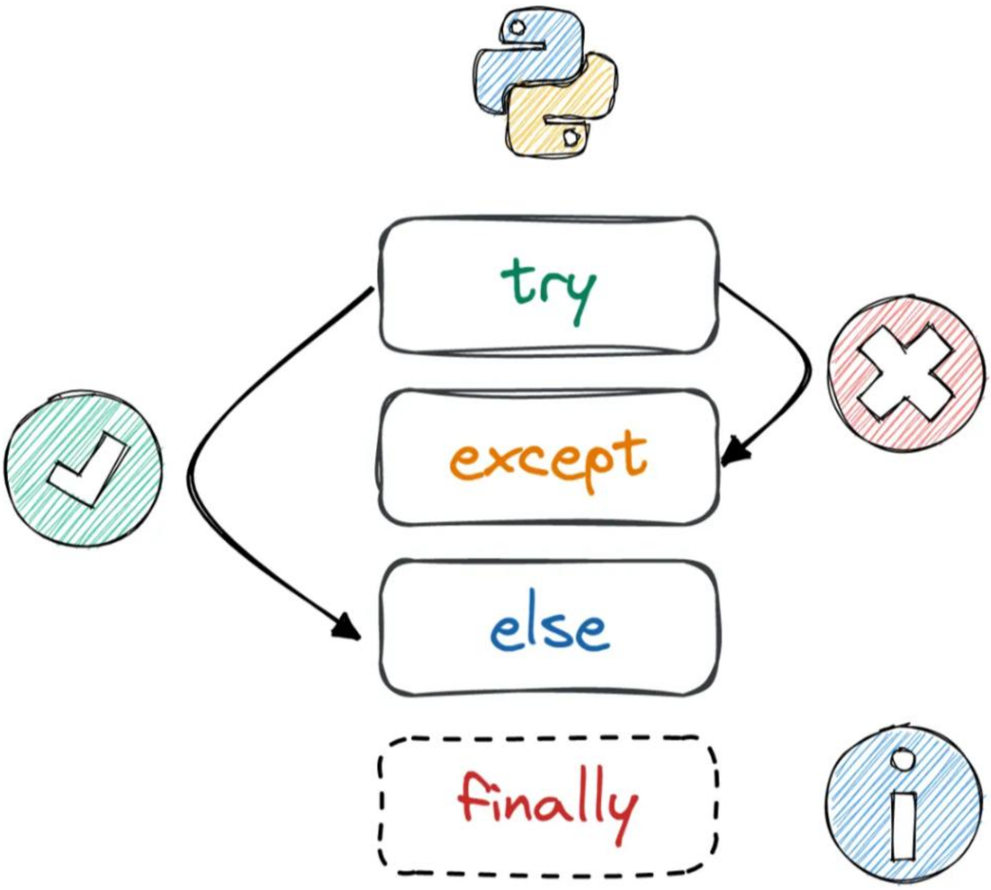
Explain Exception Handling with an example



Exception Handling: The "Emergency Plan"

Imagine you are a **chef in a restaurant kitchen**. You have a recipe (your program's code) that tells you step-by-step how to prepare a dish. Most of the time, things go smoothly.

But sometimes, **unexpected problems** occur:

- You're supposed to get an ingredient from the fridge, but it's **empty** (like trying to open a file that doesn't exist).
- You're told to divide the ingredients into 4 equal parts, but someone accidentally tells you to divide it into **0 parts** (like trying to divide by zero).
- You expect a numerical measurement, but someone gives you "five hundred grams" written in **text** instead of the number 500 (like getting text when you expect a number).

If you, the chef, don't have a plan for these unexpected problems, you might just **stop cooking altogether** and yell, "ERROR! Cannot proceed!" (This is like your program crashing).

Exception Handling is like having an **"Emergency Plan"** for your program. It's a set of instructions that tells your program: "Hey, try to do this normal operation. BUT, if something unexpected (an 'exception') happens, don't just crash! Instead, follow *these backup steps* to handle the problem gracefully."

How it Works in Python:

Python uses try, except, and optionally else and finally blocks:

- **try:** You put the code that *might* cause a problem inside this block. It's like saying, "Try to do this step in the recipe."
- **except:** If an error (an "exception") occurs in the try block, the program immediately jumps to the except block. You can specify different except blocks for different types of errors. It's like saying, "IF the fridge is empty, THEN go check the pantry instead."
- **else (Optional):** This block runs ONLY if the code inside the try block executed **without any errors**. It's like saying, "IF everything went fine in the main cooking step, THEN proceed to plating."
- **finally (Optional):** This block always runs, no matter what happens in the try or except blocks. It's like saying, "Regardless of whether the cooking worked or not, ALWAYS clean up the kitchen."