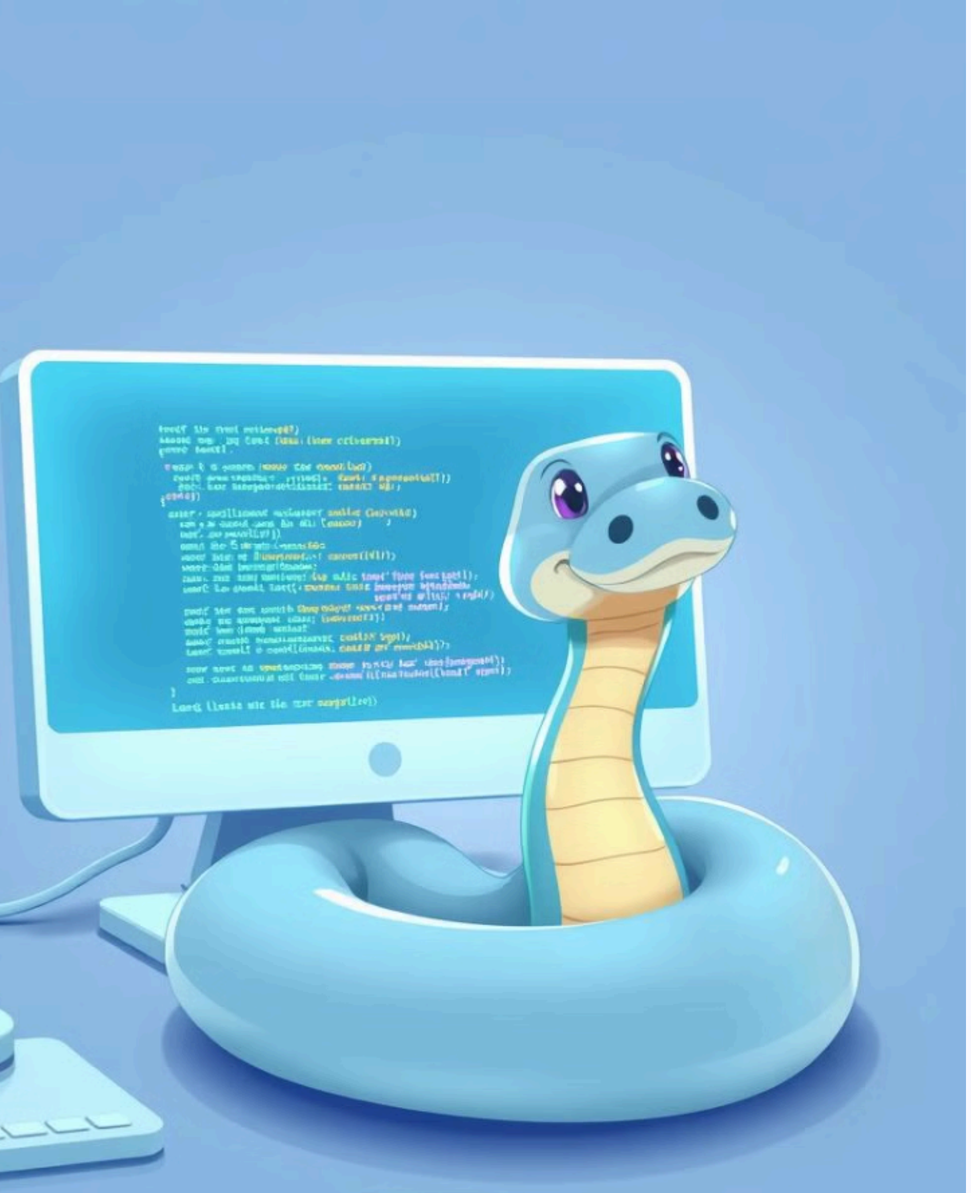
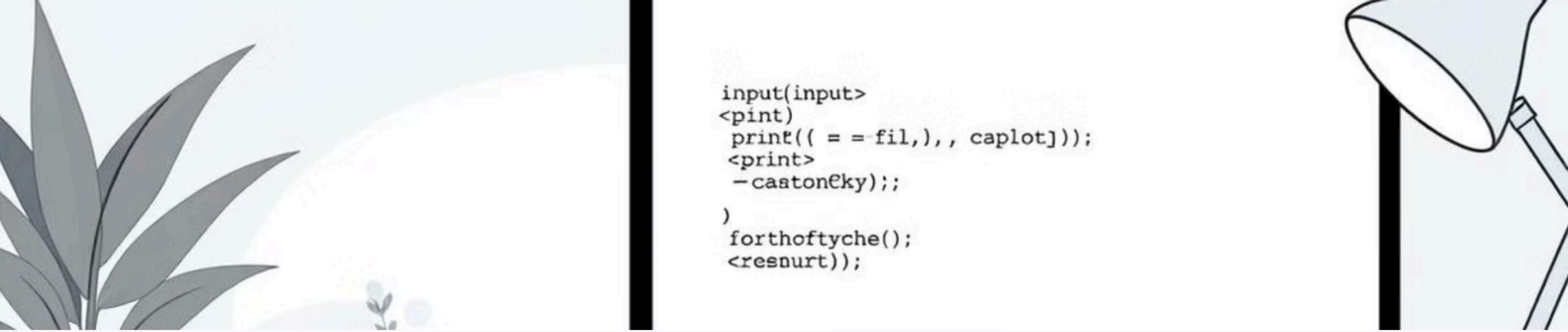


User Input and Output Basics in Python

Day 7 of 75 Days Coding Challenge.

Master Python's interactive capabilities.

by Aravind K



Introduction to User Input and Output

1 **Input()**

Capture user data effortlessly.

2 **Print()**

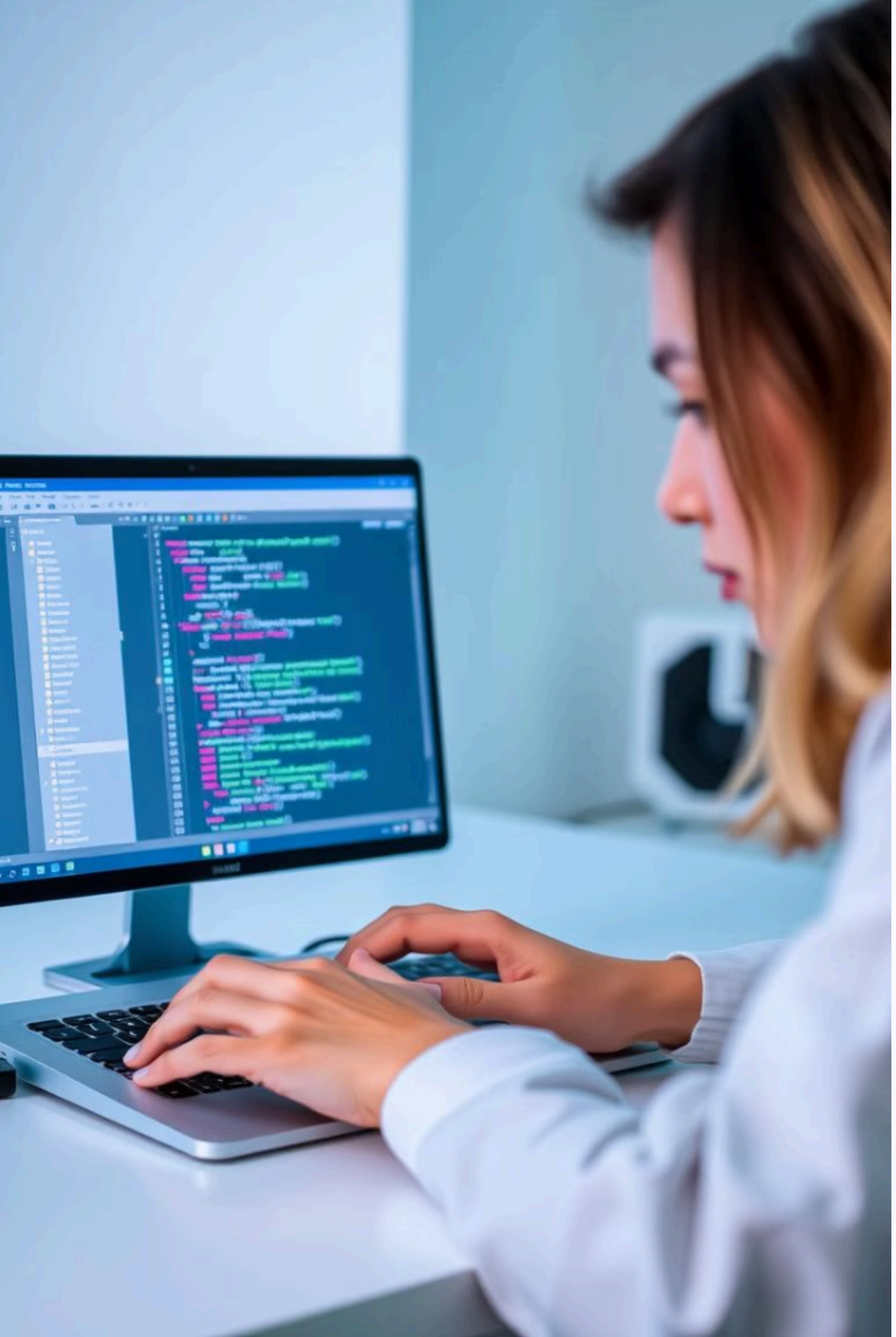
Display information clearly and concisely.

3 **F-strings**

Format output for enhanced readability.

4 **Error Handling**

Gracefully manage unexpected user inputs.



Taking Input from Users

1

String Input

Capture text with input() function.

2

Numeric Input

Convert strings to numbers using int() or float().

3

Multiple Inputs

Gather various data types in sequence.

Displaying Output with print()

Basic Syntax

print() displays values to console.

Multiple Arguments

Separate values with commas for easy formatting.

Customization

Modify separators and line endings for precise output.

```
feetpree-e0... >
 batates fill, rodent, coston,
 Mateline (ft, the firrent.
 Vatalibee ll) esment,
 Tapercaste alire, irtetent,
 batorlbesfit, to fobceltbrohtping,
 Raterible it, lout, reimenated.<
 Vatorible in, to, ft, inbo tn taltt,
 Viyatlbeles, incouel,
 Watortbee thı roe-ic, respore(])

 Rebrite fill, instome teste,
 <<Zuntheales, talit,
 <tystibeles, f..t.
 Uatitanbles (biit filt Pictigurg.
 Betorre aire, to (outfoe,/virementers,irclls,
 Beteling fit, toser fritthomeschittl,
 Matoribe ill; ratlof struigg,
 Matertble ft, tuira, srsud,;
 Mocaribee itt. bitc./frtatoner.bitty,
 Matoribe fit, toses yecter self)
 Matortbellt.- toset (frtterbour fill, irc-isall)
 Vatabibes:(fral irtt, frtyter scctes,
 Watoribee for the it. tacuct;
 eod.
 Astoltables - asset fratatning;
 Vatattbele slures (istericcAbett.(rait i.ll))


 Matoribee fill >
```

String Formatting with f-strings

Syntax

Prefix string with 'f' and use {}.

Variable Insertion

Place variables directly inside curly braces.

Expressions

Evaluate expressions within f-string brackets.

Formatting Options

Specify alignment, padding, and precision easily.

Handling Input Errors

1

Try Block

Attempt to execute potentially error-prone code.

2

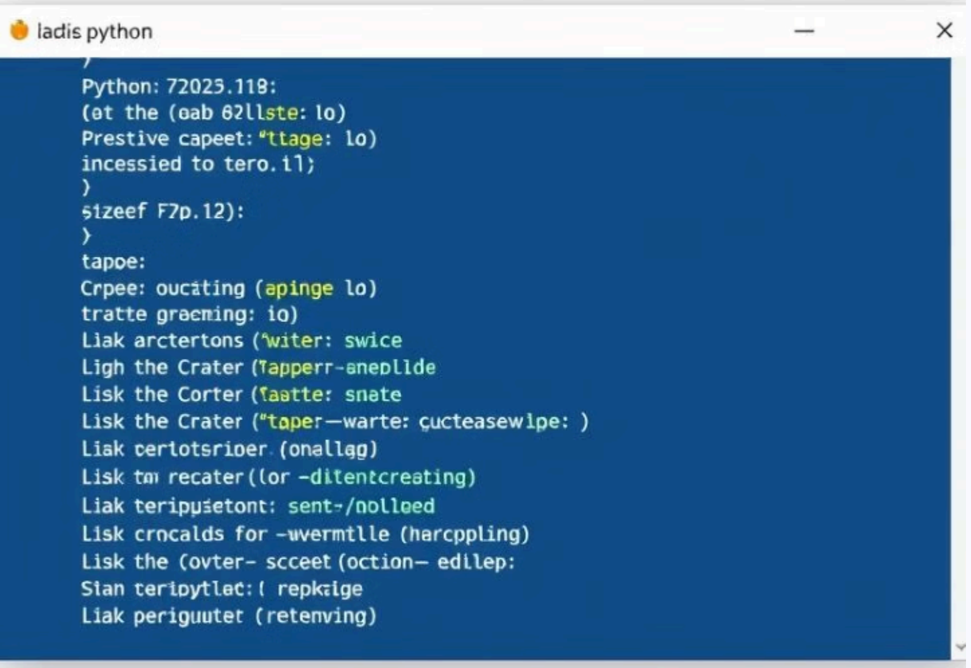
Except Block

Catch and handle specific exceptions gracefully.

3

User Feedback

Provide clear error messages for better interaction.



Putting It All Together

Input	Processing	Output
Name (string)	Store as-is	Formatted greeting
Age (integer)	Convert and validate	Age confirmation
Invalid input	Error handling	Friendly error message

Conclusion

User Input

Capture data effectively with input().

Output Display

Present information clearly using print().

Formatting

Enhance readability with f-strings.

Error Handling

Ensure smooth execution with try-except.