



Input and Output: The Foundation of Programming

This presentation defines input and output. It explores the crucial role of I/O in program design. Programs act as black boxes, using I/O for communication.

What is Input?

Data Source

Input represents data programs receive. This information comes from external sources.

Input Variety

Sources include keyboards, files, networks, and sensors. Data types vary, including text, numbers, images, audio, and video.

Real World Input

Consider a user entering a name. They enter it into a registration form.



```

21 buctiler, swlese, apciafle, "List_havalline-, Fuhectt, interffit, enuredal -> you really want to know
13 list (culture for_coclegyel)}
29 t= fouthelitter_raylar 1112);
23 criver the lant_Perer(talner del_idt;nins;) all, congeectly (Lae("ville_sare, larew(2) -> you really want to know
24 ab= robult susedelle)) siff-eslemte1))
26 eucolunise; (datior_iod lawesfortici) clouger)) (Caw; let_ABT; -> you really want to know

```

Input Methods and Functions



Python

``input()``



Java

``Scanner.nextLine()
()``



C++

``cin >>``



JavaScript

``prompt()``

Each language has input methods. Error handling is crucial using "try...except" blocks.



What is Output?

Data Destination

Output defines how a program sends data. Output goes to external destinations.

Output Variety

Destinations encompass screens, files, printers, and networks. Data types also vary, like text, numbers, images, audio, and video.

Real World Output

An example is a result from a calculation. It can be displayed to a user.

Output Methods and Functions



Python

`print()`



Java

`System.out.println()`



C++

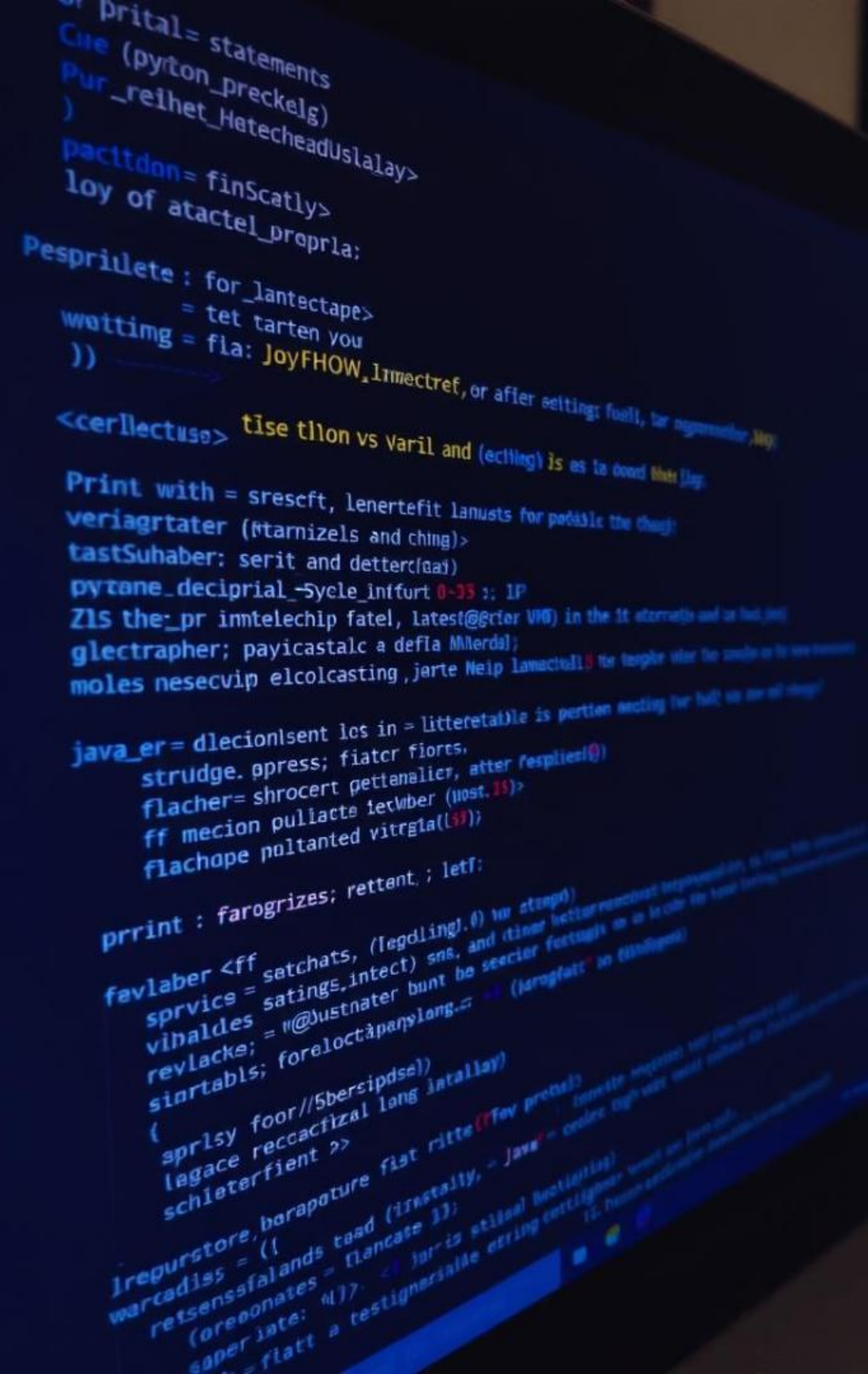
`cout <<`



JavaScript

`console.log()`

Formatting output is important. Use decimal places, alignment, and string formatting.



Input/Output Streams

1

Streams Defined

Streams are sequential channels for I/O operations.

2

STD I/O

Includes standard input, output, and error streams.

3

Stream Redirect

Redirecting stdout is common: ``program.exe > output.txt``.

4

Stream Piping

Connect output to another program's input (``program1 | program2``).



Real-World Examples

CSV Analysis

**Read data, calculate
stats (average, median).**

Sensor Display

**Receive sensor data,
show on dashboard.**

Simple Calculator

**Command-line input,
display result.**

API Data Fetch

**Read website data,
present to user.**

Best Practices & Conclusion

Validate Input

Prevent errors and security vulnerabilities.

Meaningful Output

Guide users with clear messages.

Appropriate Methods

Choose proper I/O for tasks.

User Focus

I/O bridges humans and computers. Focus on the user!

