

# CS1302 Introduction to Computer Programming

## Course overview

Prof. Weitao Xu  
[weitaoxu@cityu.edu.hk](mailto:weitaoxu@cityu.edu.hk)

# About the course

This course aims to introduce key concepts, techniques, and good practices of programming using a high-level programming language Python.

Course Intended Learning Outcomes (CILOs):

1. Explain the structure of a computer program.
2. Analyze, test and debug computer programs.
3. Apply proper programming techniques to solve a task.
4. Construct well-structured programs.

# About the course

- **Lecturers**

Dr. Chung Chan, [chung.chan@cityu.edu.hk](mailto:chung.chan@cityu.edu.hk) (Course leader, Course management)

Dr. Weitao Xu [weitaoxu@cityu.edu.hk](mailto:weitaoxu@cityu.edu.hk) (Teaching)

Dr. Helena Wong, [cshwong@cityu.edu.hk](mailto:cshwong@cityu.edu.hk) (Teaching)

- **TAs**

- We have about 20 TAs to offer assistance on teaching and QA sessions.

- **Teaching mode**

- Lecture : On-campus face-to-face
  - Labs: On-campus face-to-face

Lectures/Labs will be recorded on Zoom, but the recording is used for review purpose. You're encouraged to attend the class physically.

Attendance will not be marked, but individual lectures may record attendance for progress check.

# Assessment

- **Coursework (50%) :**
  1. Assignments: 30% = E-Quizzes (24%)+CE-Quizzes (6%)
  2. Midterm Quiz: 20%
    - Date: 28-Feb-2026 (Saturday) 18:30-19:30, mark it on your calander
    - Computer-based: you will use Safe Exam Browser on the provided computers to access the exam (without connection to Internet during exam except Canvas).
- **Final Exam (50%)**

The final exam will be on computer, and the duration is 2 hours.

## Passing criteria:

At least 30% of the maximum mark for the examination must be obtained.

- The distribution of scores is different every year, so it's hard to tell the boundaries of A+/A/A-/B+/B/B-.
- The final grade is determined by the relative position of your score (top 3%, 20%, 50%) but not the absolute value of your score (90, 85, 70..)

# Course materials

- Course contents are delivered in the Jupyterhub platform

2025/26 Semester B

- Home
- Announcements
- People
- Assignments
- Zoom
- Grades
- Files
- LOQ
- Discussions
- Quizzes
- Pages
- Syllabus
- Modules
- Collaborations
- Outcomes
- Rubrics
- BigBlueButton
- Settings

CE-Quiz 0

Completion

1. Introduction to Computer Programming

Lecture

- Introduction to Computer Programming

Reading

- [Haltermann17] 1.1-1.3, 2.1-2.3

Assignments

E-Quiz 1

Completion

CE-Quiz 1

Completion

- Textbook

Haltermann, R. L. "Fundamentals of Python programming." (2017)

- [E-Book available from openlibrary.org \(Links to an external site.\)](#)
- [Source code available from GitHub repository](#)

# Course materials

1. Introduction to Computer Programming

Completion ▾

**Lecture**

- Introduction to Computer Programming

Completion ▾

**Reading**

- [Halterman17] 1.1-1.3, 2.1-2.3

Completion ▾

**Assignments**

 E-Quiz 1

 CE-Quiz 1

The window is too small?  
Try to right-click, and “Open link in new tab”

Assignments

 E-Quiz 1

 CE-Quiz 1

- Open link in new tab
- Open link in split view
- Open link in new window
- Open link in incognito window
- Save link as...
- Copy link address

Courses

# E-Quiz

## E-Quiz 1

Receive a grade

**Opens:** Monday, 19 January 2026, 12:00 AM

**Closes:** Sunday, 25 January 2026, 11:59 PM

### Instructions

This is an **iRAT** (Individual Readiness Assurance Test) that assesses your personal understanding of the lecture material.

### Timing

- The **opening and closing times** are displayed on this cover page.
- Your answers are **automatically submitted** at the closing time, after which you can review your scores and answers.
- If you click **Finish attempt...** and then **Submit all and finish**, you will no longer be able to view or modify your answers.

### Answer Checking Options

Each question may include one or both of the following buttons:

#### Check Button

- Allows you to verify your answer **before submission**.
- **Penalty applies** for incorrect attempts, E.g., you may see
  - **10%** for programming questions; and
  - **100%** for True/False questions.
- Designed to discourage random guessing.
- For programming questions, there are **hidden test cases** that are evaluated only after submission to prevent test-specific solutions.

#### Precheck Button (Programming Questions Only)

- Lets you test your solution against a **smaller set of test cases** than what the **Check** button does.
- **No penalty** is applied.

Read the instructions carefully

# CE-Quiz

## CE-Quiz 1

Receive a grade

**Opens:** Monday, 26 January 2026, 12:00 AM

**Closes:** Wednesday, 28 January 2026, 11:59 PM

### CE-Quiz Instructions

The **CE-Quiz** (Collaborative E-Quiz) assesses your understanding of the relevant topics in a **Collaborative Mode with Socratic AI**.

- Questions are mostly selected from E-Quiz questions.
- You must take the CE-Quiz **on site** at the venue for your assigned section during its scheduled time.
- Similar to the closed-book midterm and examination, students must use the **Safe Exam Browser on a Lab PC** (NOT students' laptops) to do the test. Other electronic devices such as smart glasses and mobile phones are not allowed. Forbidden devices must be placed inside the bag or at the front of the venue.
- While collaboration is allowed for CE-Quizzes, you may
  - use **OPTMentor**, an AI-assisted Visual Debugger;
  - seek help from classmates, student helpers and TAs in the form of **hints but not answers** .

Read the instructions carefully

### Timing

- The assessment is **only available during the Lab period**.
- Your answers are **automatically submitted** at the closing time.
- You can **view your score** and **review your attempt** from this cover page after the closing time.
- If you click **Finish attempt...** and then **Submit all and finish**, you will no longer be able to view or modify your answers.

### Answer Checking Options

Each question may include one or both of the following buttons:

#### ◆ **Check** Button

- Allows you to verify your answer **before submission**.
- **Penalty applies** for incorrect attempts, E.g., you may see
  - 10% for programming questions.
- **Designed to discourage random guessing**

Academic Honesty  
is regarded as a serious academic  
offence in the University.

Do assignments and exercises **on your own**

**“On your own” means**

- ✓ discuss the problems with any other people.
- ✓ study materials available on the internet.
- ✓ refer to books.



**But the details and write-up must be entirely your work.**

The **principle** is: **Students should gain through  
practicing and developing skills in doing your work.**

# Academic Honesty

is regarded as a serious academic offence in the University.

Do assignments and exercises **on your own**

Deserved mark?

Unfair situation ✗

You should not create any chance for other students to copy your work.

For any plagiarism case,

–The student who plagiarizes will be punished.

–Any student who allows his/her work to be copied will **also** be punished.

# Do students in the following cases demonstrate academic honesty?

## Case 1

- Chi Wai did some research on the Internet and found some useful sources for his assignment. He used the ideas from these sources but forgot to cite the sources in the text of his assignment and provide a reference list.

## Case 2

- Mia searched the Internet and found a useful diagram. She modified the diagram and included it in her assignment, but did not cite and reference the source.

## Case 3

- Nick handed in the same assignment for more than one subject/course.

## Case 4

- Xiaobo copied his assignment from a classmate's work, put his own name on it and submitted it to his teacher.

## Case 5

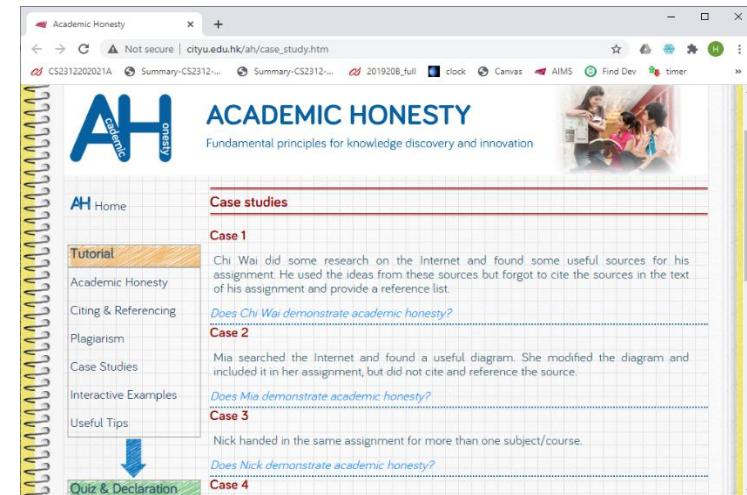
- Kit gave Suki his assignment and Suki copied it, put her own name on it, and submitted it to her teacher.

## Case 6

- Chloe and Eason worked on an individual assignment together. They each did a different part of the assignment. Then, they shared their work, put their own name on the assignment and individually submitted the work to their teacher.

## Case 7

- Ming and Doris discussed some ideas about how to do an individual assignment. Then, they completed the assignment on their own, without looking at each other's work. They put only their own name on their assignment and individually submitted their work to their teacher.

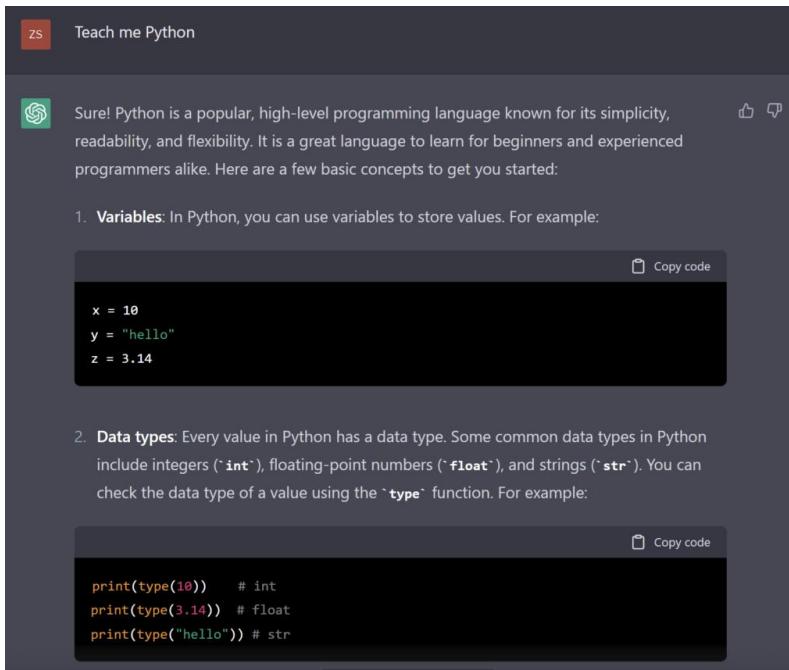


The screenshot shows a web browser window for 'Academic Honesty' with the URL [https://www.cityu.edu.hk/ah/case\\_study.htm](https://www.cityu.edu.hk/ah/case_study.htm). The page features a spiral notebook icon on the left and a main content area with a blue header 'ACADEMIC HONESTY' and sub-header 'Fundamental principles for knowledge discovery and innovation'. It includes a photo of students studying. The left sidebar has a 'Tutorial' section with 'AH Home', 'Case studies', 'Case 1', 'Case 2', 'Case 3', and 'Case 4'. The 'Case 1' section discusses Chi Wai's actions. The 'Case 2' section discusses Mia's actions. The 'Case 3' section discusses Nick's actions. The 'Case 4' section discusses the shared assignment between Chloe and Eason. A blue arrow points from the 'Case 1' section to the 'Case 2' section.

<https://www.cityu.edu.hk/ah/Tutorial/casestudy>

# Can I use ChatGPT

- You're free to use any tools that can help your study, but you won't have access to Internet during the midterm/final exams



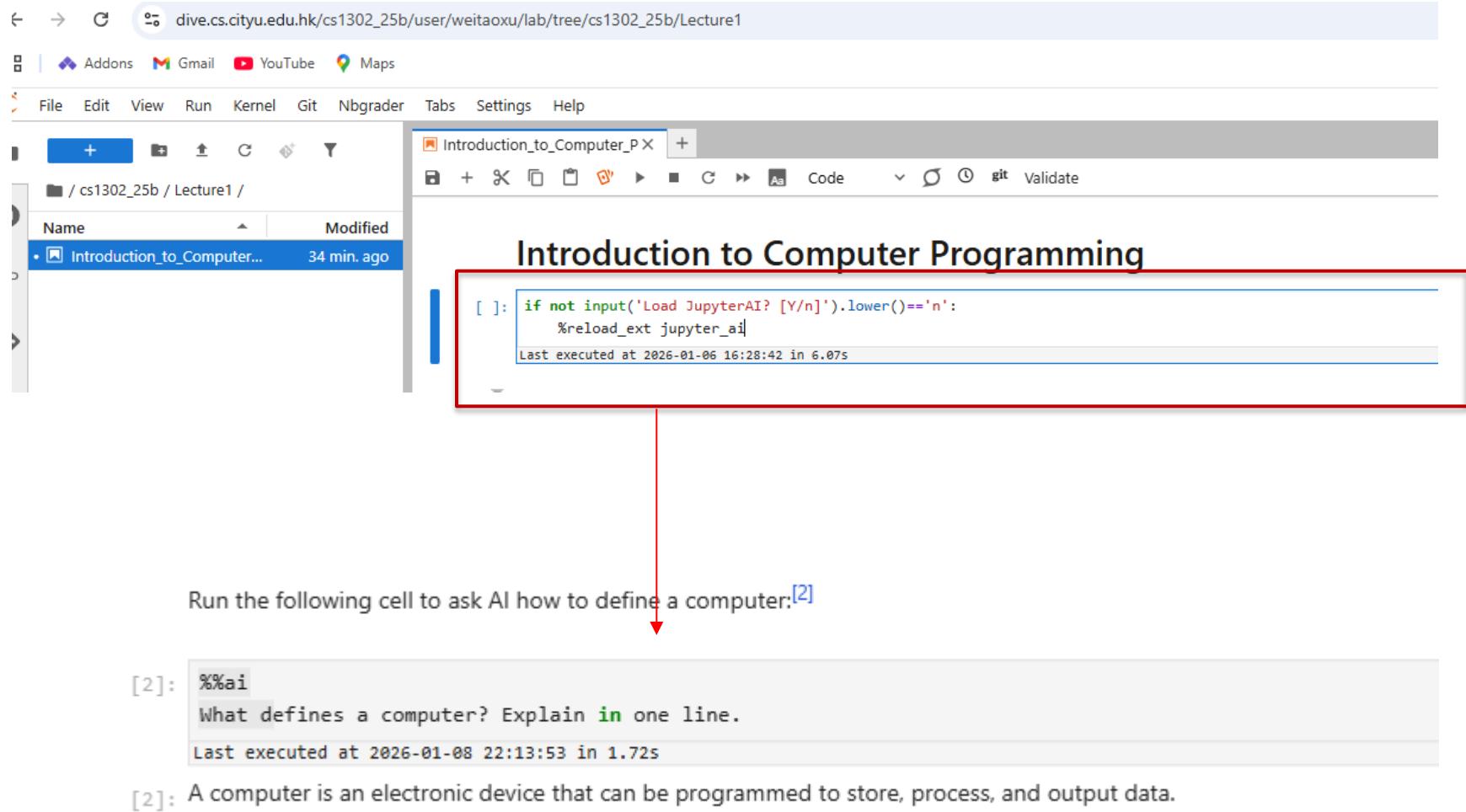
The image is a screenshot of a digital interface, likely a chatbot or AI tool, designed to teach Python. The interface has a dark background with light-colored text and code snippets. At the top, there is a small orange box with the letters 'zs' and the text 'Teach me Python'. Below this, a green icon of a robot head is followed by a message: 'Sure! Python is a popular, high-level programming language known for its simplicity, readability, and flexibility. It is a great language to learn for beginners and experienced programmers alike. Here are a few basic concepts to get you started:'. The first concept is 'Variables', with an example code block:

```
x = 10
y = "hello"
z = 3.14
```

Below this, another concept is 'Data types', with an example code block:

```
print(type(10))    # int
print(type(3.14))  # float
print(type("hello")) # str
```

# Generative AI is also available in the notebooks



The screenshot shows a Jupyter Notebook interface with a red box highlighting a cell in the 'Introduction\_to\_Computer\_Programming' notebook. The cell contains the following Python code:

```
[ ]: if not input('Load JupyterAI? [Y/n]').lower()=='n':  
    %reload_ext jupyter_ai
```

Below this, a red arrow points down to another cell in the same notebook, which contains the command `%%ai`. The output of this cell is:

Run the following cell to ask AI how to define a computer.<sup>[2]</sup>

```
[2]: %%ai  
What defines a computer? Explain in one line.  
Last executed at 2026-01-08 22:13:53 in 1.72s
```

The final output of the cell is:

```
[2]: A computer is an electronic device that can be programmed to store, process, and output data.
```

# About the course – Key to Success

## Practice with understanding

**The solution is not everything.**

You need to learn **the approach** to build the solution step by step.

## Learn from errors (**debugging strategies**)

Mistakes are often **good teachers** 

## Digest well to learn the 100%

Memorizing and dictation,  
if without understanding,  
are worthless,  
especially in a programming course!

# Python is a tool to solve questions

In some courses, what you learn in the class

$$x^2 + 2 = 18$$

$$x^2 = 18 - 2$$

$$x = 4$$

similar

$$x^2 + 9 = 100$$

$$x^2 = 100 - 9$$

$$x = 8$$

What you see in the exam

In this course, what you learn in the class

## Python for loop

A for loop is used for iterating over a sequence and iterables (like range, list, a tuple, a dictionary, a set, or a string).

Indentation  
Loop body is must be properly indented

```
for i in range(5):  
    statement 1  
    statement 2  
    ...  
    statement n  
else:  
    statement(s)
```

Definite iterations.  
(Total 5 iterations)

Body of for loop  
Execute till the last item of a sequence

Else Block (optional)  
Execute only when for loop executes normally

unseen

What you see in the exam

## Example 1: Program to print half pyramid using \*

```
*  
* *  
* * *  
* * * *  
* * * * *
```

## Source Code

```
rows = int(input("Enter number of rows: "))  
  
for i in range(rows):  
    for j in range(i+1):  
        print("* ", end="")  
    print()
```

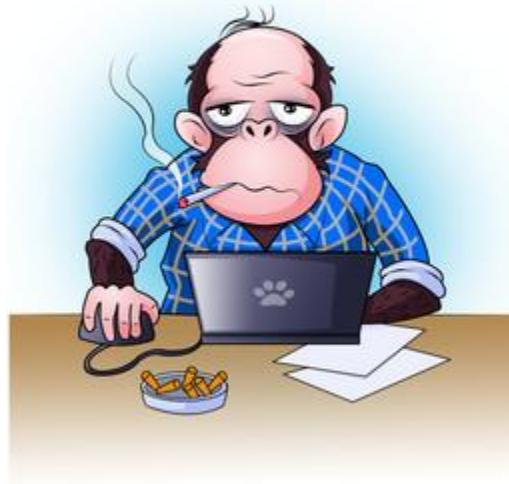
# After this course

Become a Programmer

Cool and fun

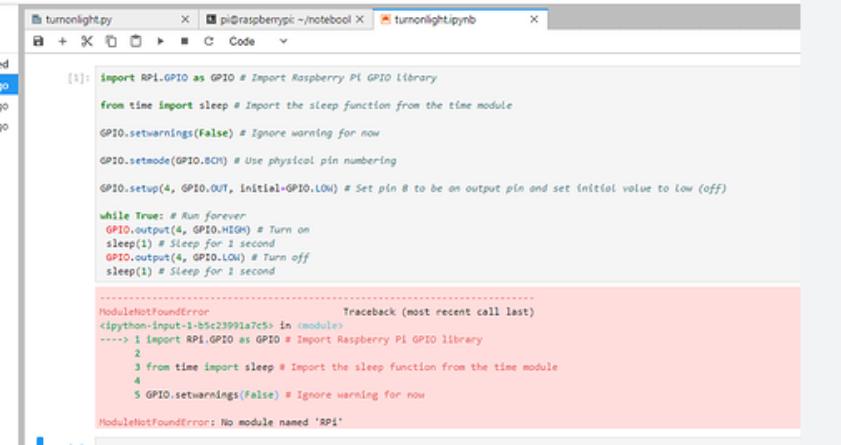
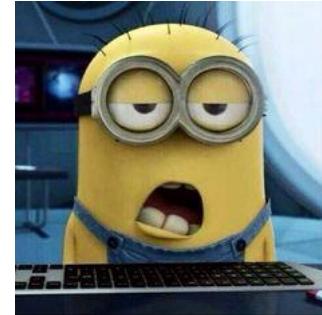


# But being a programmer (程序猿)...



# What to expect

- Programming is fun
- *Learning programming may not!*
- You'll see lots of syntax (rules for programming languages), yes they may look boring...
- You'll see lots of errors, that'll take you lots of time to fix...



```
[1]: import RPi.GPIO as GPIO # Import Raspberry Pi GPIO library
from time import sleep # Import the sleep function from the time module
GPIO.setwarnings(False) # Ignore warning for now
GPIO.setmode(GPIO.BCM) # Use physical pin numbering
GPIO.setup(4, GPIO.OUT, initial=GPIO.LOW) # Set pin 8 to be an output pin and set initial value to low (off)

while True: # Run forever
    GPIO.output(4, GPIO.HIGH) # Turn on
    sleep(1) # Sleep for 1 second
    GPIO.output(4, GPIO.LOW) # Turn off
    sleep(1) # Sleep for 1 second

ModuleNotFoundError: Traceback (most recent call last)
<ipython-input-1-b5c23991a7c5> in <module>
----> 1 import RPi.GPIO as GPIO # Import Raspberry Pi GPIO library
      2
      3 from time import sleep # Import the sleep function from the time module
      4
      5 GPIO.setwarnings(False) # Ignore warning for now

ModuleNotFoundError: No module named 'RPi'
```

# What to expect

- To have fun, usually we need to go through lots of detail stuff first -> learning curve

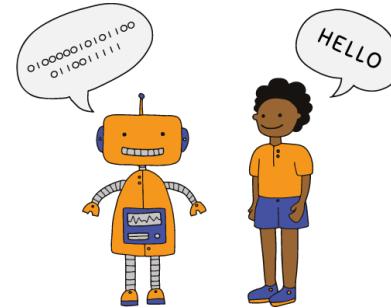
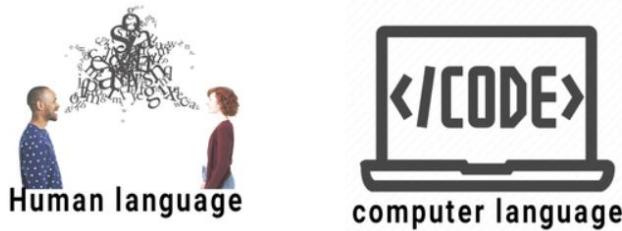


## Positive thinking

- You learn a language .. What are the stories behind the details? You *communicate* with the language.
- You create solutions and solve mistakes
  - You are winning the games!

# One more tip

- If the computer says you're wrong, you are wrong!  
Don't doubt the computers, ever. They are always right  
(unlike CNN or Trump...)



- Focus on finding out what make the computer think you're wrong
  - Debug
  - Why are they called bugs?



# There are many Programming Languages in the world

Ada Assembly Basic **C C++ C#** Cobol

Cobra CODE ColdFusion Delphi Eiffel

Fortran FoxPro GPSS J# J++ **Java**

**JavaScript LISP** Logo LUA MEL Modula-2

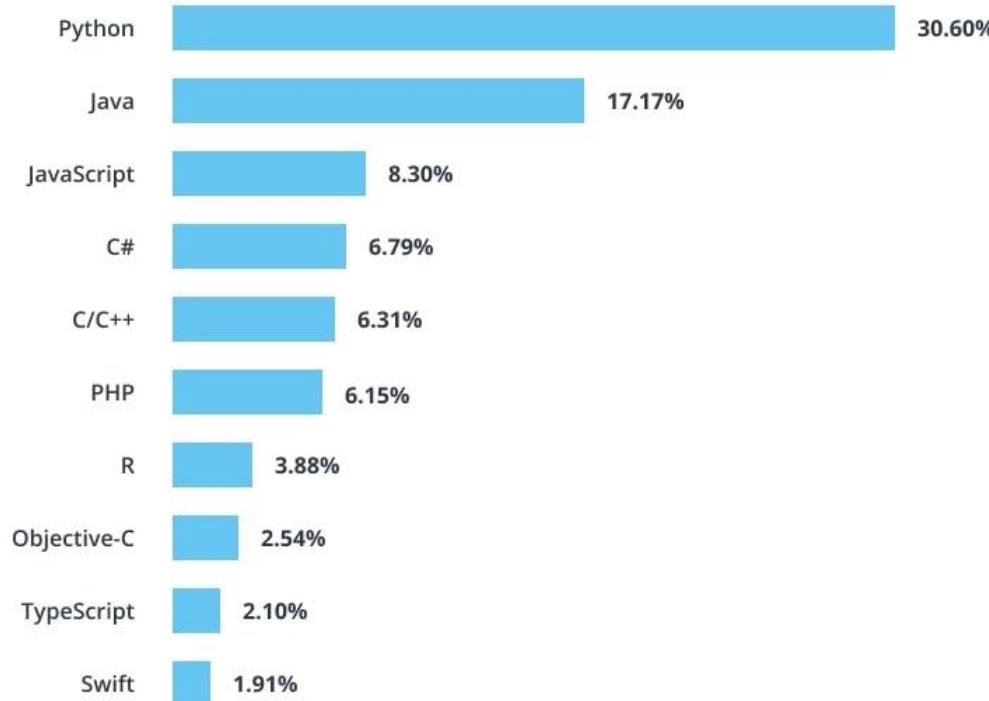
Miranda Objective-C **Perl PHP** Prolog

**Python Go** Visual Basic Swift

# Programming Language

Python:

The most popular programming language



# Programming Language

...

What programming language is mostly used in the area of AI? Why?



GPT-3.5-Turbo

Poe

...

Python is the programming language that is predominantly used in the field of AI. There are several reasons for this:

- 1. Simplicity and Readability:** Python is known for its simplicity and readability. Its syntax is clear and concise, making it easier for developers to write and maintain code.
- 2. Rich Ecosystem:** Python has a vast ecosystem of libraries and frameworks that are specifically designed for machine learning and AI, such as TensorFlow, PyTorch, and scikit-learn. These libraries provide tools for tasks like data manipulation, mathematical operations, and model building.
- 3. Community Support:** Python has a large and active community of developers. This means that there are plenty of resources, tutorials, and forums available for AI practitioners to seek help and share knowledge.
- 4. Flexibility:** Python is a versatile language that can be used for a wide range of applications beyond AI. This flexibility makes it a popular choice for developers who work on diverse projects.
- 5. Performance:** While Python may not be the fastest language for certain tasks, its performance can be optimized using libraries like NumPy and by leveraging its integration with high-performance languages like C and C++.

Overall, Python's ease of use, extensive libraries, community support, and flexibility make it the preferred choice for AI and machine learning applications.

# Programming Language

Almost all the open-sourced AI projects are built upon **Python**.

For example:

## 1. Segment Anything

- <https://segment-anything.com/demo>

## 2. Stable Diffusion

- <https://huggingface.co/spaces/stabilityai/stable-diffusion-3-medium>

## 3. Music Generation

- <https://musicgeneratorai.com/>

## 4. Pose Estimation

- <https://saiwa.ai/app/detection/pose-estimation-1/>

## Average Python Developer Salaries

Switzerland	\$133,000
The USA	\$127,000
Denmark	\$81,800
Israel	\$71,000
Norway	\$62,000
Czech	\$34,122
Poland	\$21,552
Hungary	\$16,771
Ukraine	\$15,906



## Look for more practice?

- There are plenty of online programming resources, but you need to choose the one suitable for this course.
- <https://www.geeksforgeeks.org/python-exercises-practice-questions-and-solutions/?ref=shm>
- <https://www.placementpreparation.io/mcq/python/>
- <https://pynative.com/python-exercises-with-solutions/>
- <https://www.w3resource.com/python-exercises/>
- <https://pynative.com/python-basic-exercise-for-beginners/>
- Or ask ChatGPT to create some programming questions for you.