

Coding Foundations & Why AI Matters

CU Boulder AI Club Workshop Series

Tate Middleton - 9/17/25



Outlook

Goals:

- Demystify + learn how to code in Python
- Understand foundations of artificial intelligence (AI) and machine learning (ML)
- Understand the state of the art (LLMs) and the future (world models, neuromorphic computing, etc!)

Format:

- Slides during workshop times, group work after (9/16, 10/14, 10/28, 11/11, 12/9)
- Coding practice during off weeks

Why should we care?

Coding:

- Coding can automate almost anything
- Can do math, control hardware, websites, and more
- AI is made from math and code!
- Saves us manual labor -> time

AI/ML:

- Can find connections that humans would never see (ML)
- Can assist in tasks like financial forecasting, security, and fraud detection
- Saves us mental labor (AI)

What is Coding?

- Simply: instructions for computers
- Computers are dumb -> We need to be VERY clear
- **Logic** (code) + **Data** (inputs from us) = a **computation**

Example:

`print()` + "Hello AI Club!" =

`print("Hello AI Club!")`



Logic (python)

Data (me)

Our computation ... lets run this!

Variables and Types

- **Variable:** a unique “codename” for information we want to store
- **Type:** information about our variable
 - **String** (text) – **str**: “Hello”, name=“Tate”
 - **Integer** – **int**: 2025, num=10
 - **Decimals/Floating point numbers** – **float**: 1.25, pi = 3.1415

“x” is now of
type int



```
x=1
```



Assign a variable to a value with “=”

Conditionals

- The “**if**” statement checks whether something is true or not, handles the true case
- “**else**” statement takes control if the “if” statement returns false
- “**else if**”, (**elif** in python), a statement between **if** and **else**, an extra check

What will
this code
output?

```
if 1+1 == 3:  
    print("1")  
elif 1+1 == 4:  
    print("2")  
else:  
    print("3")
```



“**==**” asks if
the left
side is
equivalent
to the right

Loops

- `for` loop: repeat the code in a loop for _ number of times (iterations)
- `while` loop: while _ is true, repeat the loop
- Loops are like a repeated conditional, checking something at each step!

What will these loops output?

```
steps = 10
for step in range(steps):
    print("Step:", step)
```

```
steps = 0
while steps < 10:
    print("Step:", steps)
    steps += 1
```



Python shorthand for: `steps = steps + 1`

Functions

- A **function** is a reusable block of code
- Takes inputs (as many as you want, or none), and generally has 1 output (can have more or none)

`def` means we are **defining** a function



```
def print_hello():  
    print("hello")  
  
print_hello()
```

`return` is what the function **outputs** or returns



```
def square(number):  
    return number * number  
  
output = square(3)  
print(output)
```


Off Week Practice

Make a copy of this and practice your new coding skills:

https://colab.research.google.com/drive/1X5pP_5AJ_AFpRmb4Prub1g13K-YhnkoG?usp=sharing

More Resources

- Python basics:
<https://www.youtube.com/live/fLAfa-BQtOQ>
- Feedback:
<https://forms.gle/4122RR7pzJcwrJC79>
- Lecture code:
https://colab.research.google.com/drive/1kBcSkMsbkq_tzmguU8w8QJV3KFXd_ctZ?usp=sharing
- Other resources:
<https://docs.google.com/document/d/1YaS8a6d095RFVoTop9ZsP-y1AFcW-GlE9uj7Z0K3EDo/edit?usp=sharing>

