



Introduction to Python

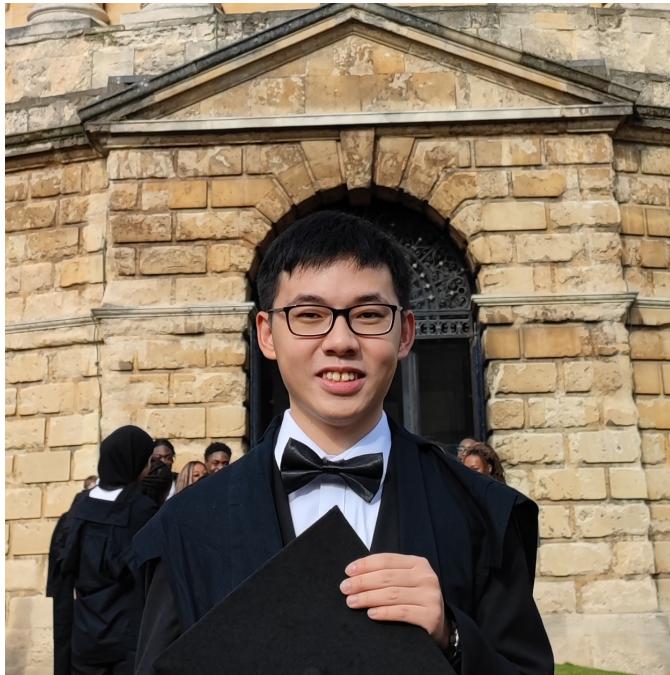
Session 1
Michaelmas Week 2

Learn to Code - Python Courses

- Complete Beginners
- Finish with coding a game
- Week 2-7 Saturdays 6pm-7pm
- Computer Science Department Lecture halls



Viktor Kozhuharov



Oscar Mui

Other events by CompSoc

- Socials on Saturdays after this session (Weeks 1-7 7pm)
- Free food if you are a member
- Talks and Workshops on Thursdays (Weeks 2-8 5pm)
- Membership fee: £1

- For more details checkout our website
- <https://ox.compsoc.net>



Where can I get the slides?

- [`https://github.com/oxcompsoc/Learntocode`](https://github.com/oxcompsoc/Learntocode)
 - ^ Session notes, slides, exercises, answers
- Also accessible from our website
[`https://ox.compsoc.net/`](https://ox.compsoc.net/)

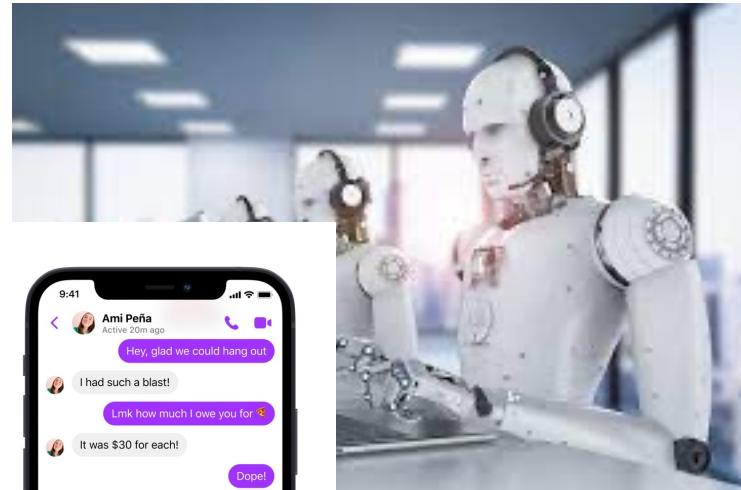
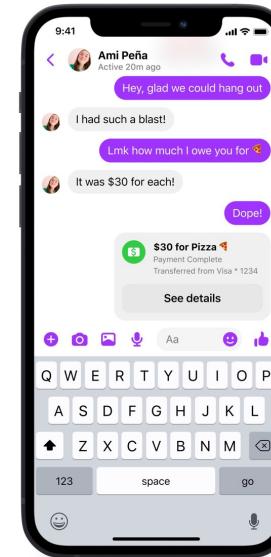
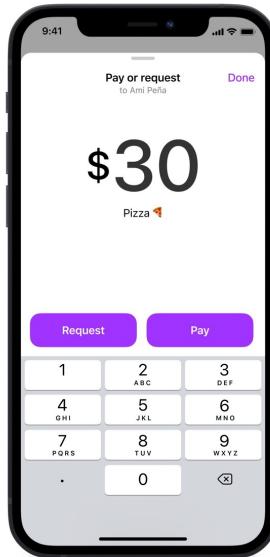
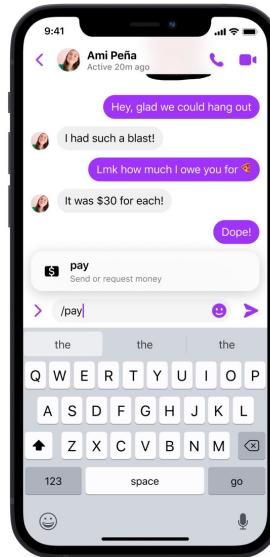
What can programs do?



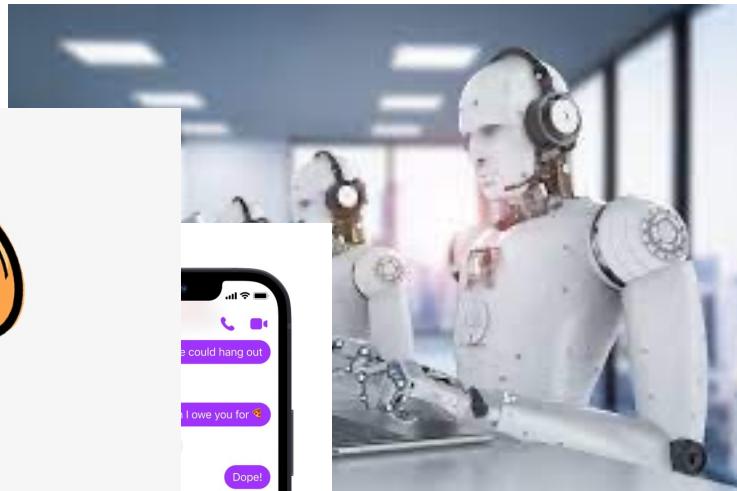
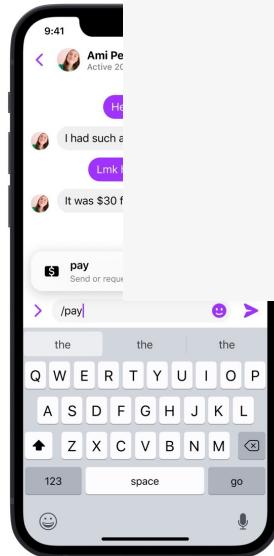
What can programs do?



What can programs do?



What can programs do?





What can WE do with Python?

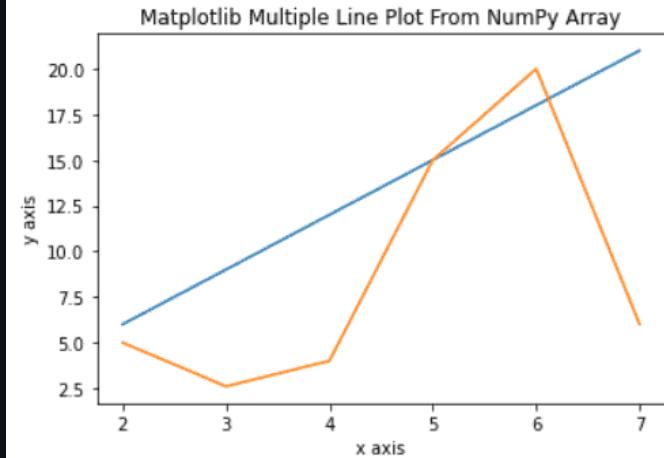
```
How many players are there? 2
Guess a number between 1 and 100
Player 1, please make a choice between 1 and 100: 50
Guess a number between 1 and 49
Player 2, please make a choice between 1 and 49: 25
Guess a number between 1 and 24
Player 1, please make a choice between 1 and 24: 12
Guess a number between 13 and 24
Player 2, please make a choice between 13 and 24: 18
Guess a number between 19 and 24
Player 1, please make a choice between 19 and 24: 22
Guess a number between 19 and 21
Player 2, please make a choice between 19 and 21: 20
Guess a number between 19 and 19
Player 1, please make a choice between 19 and 19: 19
Player 1 guessed the number! It is 19
```

What can WE do with Python?

The image shows a terminal window titled "Snake Eater" with a score of 25. A green snake is visible on the left. To the right, a code editor displays a script for a number-guessing game. The terminal output shows the game's logic, including a while loop that halves the search range until it finds the number 19.

```
rs are there? 2
between 1 and 100
se make a choice between 1 and 100: 50
between 1 and 49
se make a choice between 1 and 49: 25
between 1 and 24
se make a choice between 1 and 24: 12
between 13 and 24
se make a choice between 13 and 24: 18
between 19 and 24
se make a choice between 19 and 24: 22
between 19 and 21
se make a choice between 19 and 21: 20
Guess a number between 19 and 19
Player 1, please make a choice between 19 and 19: 19
Player 1 guessed the number! It is 19
```

What can WE do with Python?



```
between 13 and 24  
se make a choice between 13 and 24: 18  
between 19 and 24  
se make a choice between 19 and 24: 22  
between 19 and 21  
se make a choice between 19 and 21: 20
```

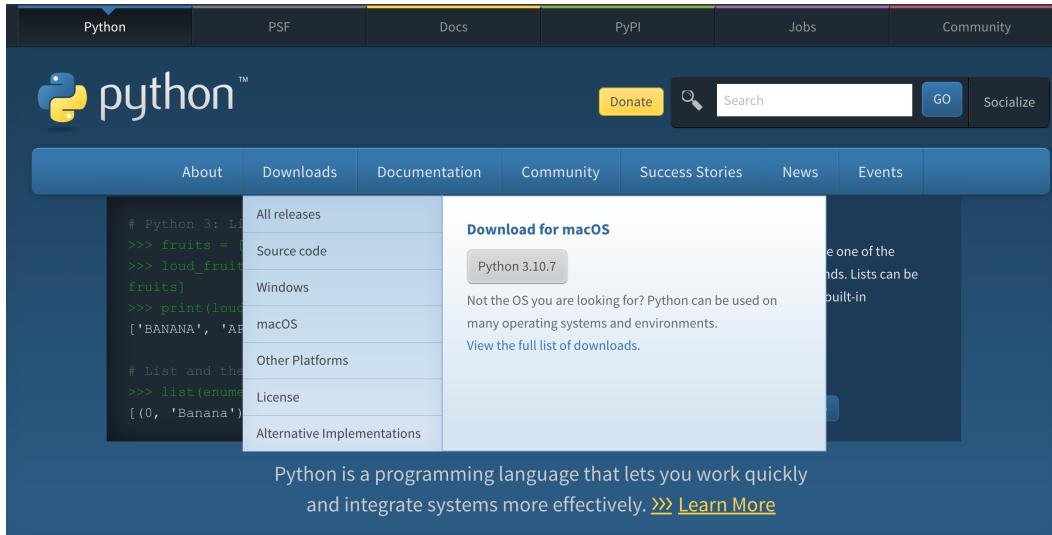
```
Guess a number between 19 and 19  
Player 1, please make a choice between 19 and 19: 19  
Player 1 guessed the number! It is 19
```

Note on pacing



Installation

- If you have any issues, you can stay behind after the session and we can help.
- For now, use repl.it.
- <https://replit.com/new/python3>



Python as a calculator

```
>>> 3+4
```

```
7
```

```
>>> 3-4
```

```
-1
```

```
>>> 3*4
```

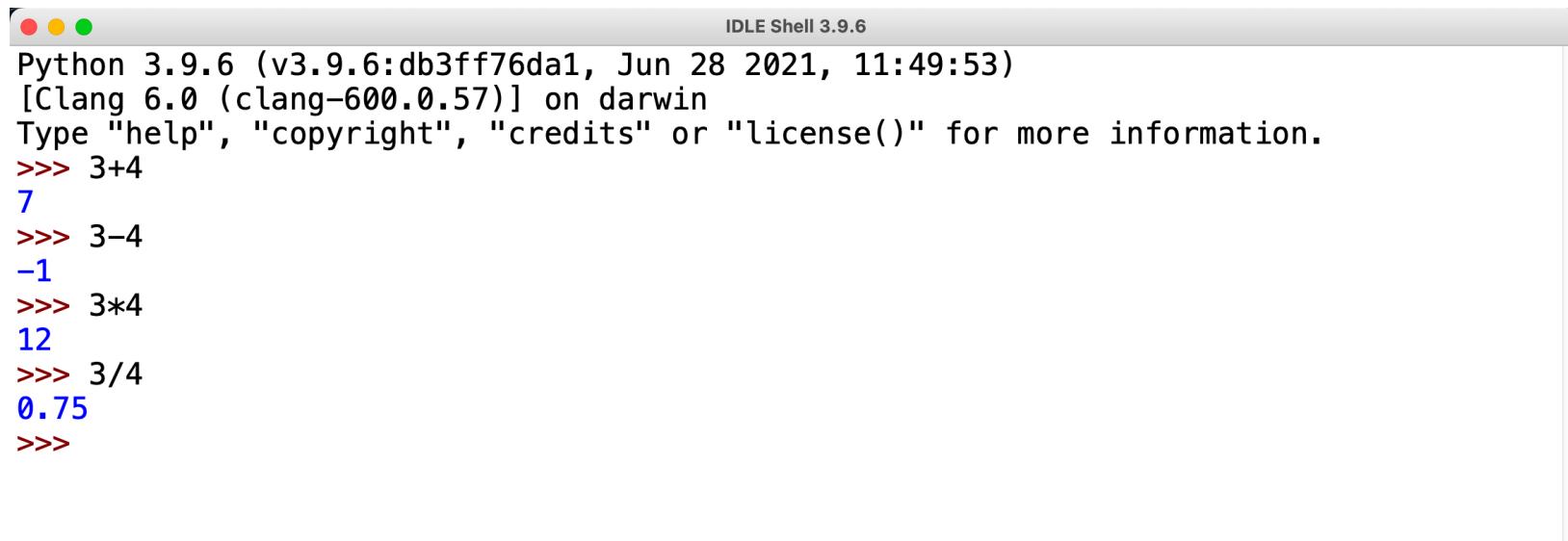
```
12
```

```
>>> 3/4
```

```
0.75
```

IDLE (Interactive Interpreter)

- Where we LEARN the basics, Line by Line



The screenshot shows the IDLE Shell 3.9.6 interface. The title bar reads "IDLE Shell 3.9.6". The main window displays the following Python session:

```
Python 3.9.6 (v3.9.6:db3ff76da1, Jun 28 2021, 11:49:53)
[Clang 6.0 (clang-600.0.57)] on darwin
Type "help", "copyright", "credits" or "license()" for more information.

>>> 3+4
7
>>> 3-4
-1
>>> 3*4
12
>>> 3/4
0.75
>>>
```

Text editor

- Where we write REAL programs

```
===== RESTART: /Users/oscar/Desktop/session1.py =====
>>>
3+4
3-4
3*4
3/4
```

You need to print them!

```
===== RESTART: /Users/oscar/Desktop/session1.py =====
7
-1
12
0.75
>>> print(3+4)
print(3-4)
print(3*4)
print(3/4)
```

Print

```
print(3+4)
```

Print text

```
print("Hello, world!")
```

Variables - as a box

```
greeting1 = "Hello!"  
my_favourite_number = 3  
pi = 3.14  
_underscore = "_"  
  
print(greeting1)  
print(my_favourite_number)  
print(pi)  
print(_underscore)
```

Variables - can be changed

```
my_favourite_number = 3
print(my_favourite_number) # Prints 3
my_favourite_number = 4
print(my_favourite_number) # Prints 4
```

Naming conventions

- Numbers, alphabets, and underscores
- Cannot start with a number
- Cannot use Python keywords (e.g. print)

Comments - Ignored by program

```
# this is the first comment
spam = 1 # and this is the second comment
# ... and now a third!
text = "# This is not a comment because it's inside
quotes."
print("Hello, world!") # Prints Hello, world!
```

Python as a calculator

```
print(3 / 4) # Division  
print(3 // 4) # Integer division  
print(3 ** 4) # Exponentiation (i.e.  
calculates 3 to the power of 4)
```

Order of Precedence - Same as normal Maths

`2 ** 2 * 3 ** 2 + 4 ** 2`

→

`((2 ** 2) * (3 ** 2)) + 4 ** 2`

Use parenthesis to change the order of precedence

Data Types

1	1.5	"a"
5	40.4550590	"b"
15	1.0	"boy"
15404505	53.4	"I am a boy!"
-455	-0.588	"Hello world"
0	-458.5	"Test"

INTEGERS

Data Types

1	1.5	"a"
5	40.4550590	"b"
15	1.0	"boy"
15404505	53.4	"I am a boy!"
-455	-0.588	"Hello world"
0	-458.5	"Test"
INTEGERS	FLOATS	
	(floating point numbers)	

Data Types

1	1.5	"a"
5	40.4550590	"b"
15	1.0	"boy"
15404505	53.4	"I am a boy!"
-455	-0.588	"Hello world"
0	-458.5	"Test"
INTEGERS	FLOATS	STRINGS
		(text)

Integers and Floats

- 4 VS 4.0
- `/` gives a float
- `//` gives an integer (rounded down)
- `print(7 / 4) # 1.75`
- `print(7 // 4) # 1 (rounded down)`
- `print(8/4) # 2.0 (a float)`
- `print(8//4) # 2 (an integer)`

Integers and Floats

- `print(4+4) # 8 (an integer)`
- `print(8.0-4) # 8.0 (a float)`
- `print(3.8*4) # 15.2 (a float)`
- `print(8**4) # 4096 (an integer)`
- `print(16**0.5) # 4.0 (a float)`

Strings

- Double quotes!!!!!!
- String "addition"
- print("Hello, " + "world!")
- String "multiplication"
- print("Hello"*3)

Type Conversion

- `int("4") # 4`
- `str(4096.5) # '4096.5'`
- `int("Hello!") # This will cause Python to crash. "Hello!" is not a number!`

Reading input

- `name = input("Please enter your name: ")`
- `print("Hi, " + name + "!")`

Example 1: Our favourite number

```
your_favourite_number = input("Please enter your favourite  
number: ")  
print("Your favourite number times two is: " + 2 *  
your_favourite_number)
```

Example 1: Our favourite number

```
your_favourite_number = int(input("Please enter your  
favourite number: "))  
print("Your favourite number times two is: " + str(2 *  
your_favourite_number))
```

Example 2: Celsius to Fahrenheit

$$^{\circ}\text{F} = 9/5 * ^{\circ}\text{C} + 32$$

Input temperature in Celsius: 25

25.0 Celsius = 77.0 Fahrenheit

Example 2: Celsius to Fahrenheit

```
c = float(input("Input temperature in Celsius: "))
f = 9 / 5.0 * c + 32
print(str(c) + " Celsius = " + str(f) + " Fahrenheit")
```

Exercises

- 1) Now try to combine the last two programs, so that you print out the person's name, as well as their favourite number multiplied by two.
- 2) Try to write a similar program that converts Fahrenheit to Celsius

$$(\text{°F} - 32) \times 5/9 = \text{°C}$$

- 3) **Average of two numbers:** create a program which takes as input two numbers and computes their average.

Where can I get the slides?

- [`https://github.com/oxcompsoc/Learntocode`](https://github.com/oxcompsoc/Learntocode)
 - ^ Session notes, slides, exercises, answers
- Also accessible from our website
[`https://ox.compsoc.net/`](https://ox.compsoc.net/)

Sneak Peak - If Statements

```
number = int(input("Please enter a number: "))  
if number < 10:  
    print(number + " is a small number.")
```

Sneak Peak - If Statements

```
number = int(input("Please enter a number: "))

if number < 10:
    print(number + " is a small number.")

else:
    print(number + " is a large number.")
```

END

Thanks for coming!