JFSS Data Science Club

PYTHON 101

Meeting #2

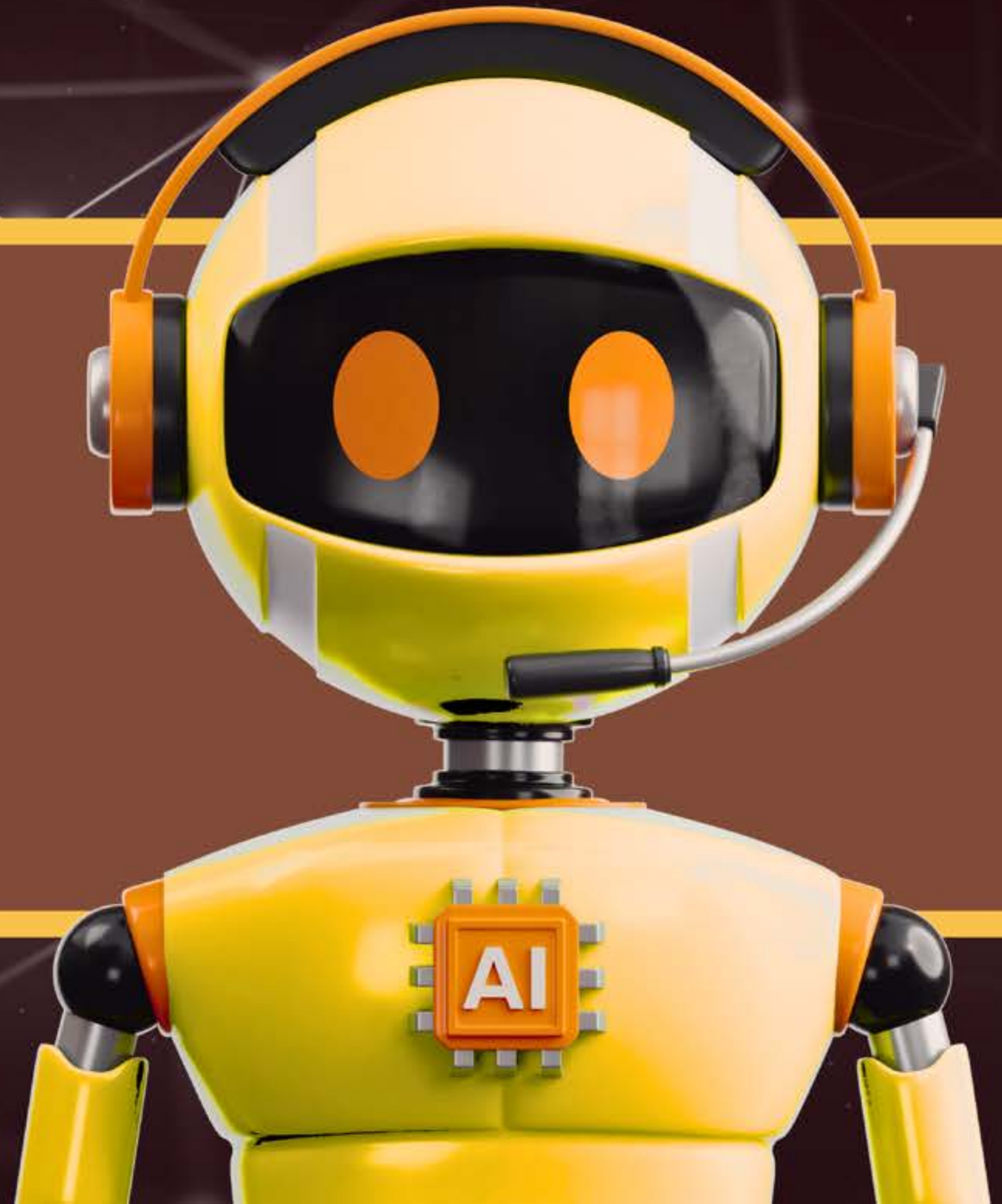


TABLE OF CONTENTS

PYTHON HISTORY

THEN

- Work began in the 1980s
- Officially released in 1991 in the Netherlands
- Python 2.0 (2000) - list comprehensions, Unicode support
- Python 3.0 (2008)

NOW

- Used in a variety of subjects
- Make algorithms
- Applicable in real-world scenarios

PRINT STATEMENT

Allows us to print to the console

```
1  print("Hello World!")
```

Console | Shell

Run 83ms on 10:13:34, 10/22

```
Hello World!
```

DATA TYPES

String

Integer

```
1  name = "Jeremy"
2
3  age = 12
4
5  likes_pineapple_on_pizza = False
6
7  weight_in_lbs = 92.7
```

Boolean

Float

VARIABLES

- Placeholders for values
- Can store different kinds of data
- No initialization required
- Data type does not need to be declared
- Data types of variables can be changed - CASTING

INPUT STATEMENT

Allows us to take in data from the user

```
1  name = input("What is your name?\n")
2  print(name)
```

```
What is your name?
Prerana   ←———— Input
Prerana   ←———— Output
```

COMMENTS

Concise notes on what the code does

(insert text here)

```
# asks for age
age = input("Please enter your age: ")
# prints age
print(age)
```

Multiline Comments

```
"""
  This comment
  can become
  more than one line.
"""
age_for_survey = input("Please enter your age: ")
```

OPERATORS

Arithmetic Operators

Comparison Operators

Logical Operators

ARITHMETIC OPERATOR

+	Addition	x + y
-	Subtraction	x - y
*	Multiplication	x * y
/	Division	x / y
**	Exponential	x ** y
//	Floor Division	x // y
%	Modulus	x % y

COMPARISON OPERATOR

==	Equal	x == y
!=	Not Equal	x != y
>	Greater than	x > y
<	Less than	x < y
>=	Greater than or equal to	x >= y
<=	Less than or equal to	x <= y

LOGICAL OPERATOR

and	Return True if both statements are true	x < 5 and x < 10
or	Return True if one statement is true	x < 5 or x < 4
not	Returns the opposite boolean	not(x < 5 and x < 10)

QUICK GAME: TRUE OR FALSE?

Fun time!!!

$x = 5$

what is the boolean value of $x < 5$?

QUICK GAME: TRUE OR FALSE?

Fun time!!!

$x = 5$

what is the boolean value of $x < 5$?

False

QUICK GAME: TRUE OR FALSE?

Fun time!!!

x = 5

what is the boolean value of x <= 5?

QUICK GAME: TRUE OR FALSE?

Fun time!!!

x = 5

what is the boolean value of x <= 5?

True

QUICK GAME: TRUE OR FALSE?

Fun time!!!

color = “red”

what is the boolean value of color == “red”?

QUICK GAME: TRUE OR FALSE?

Fun time!!!

color = “red”

what is the boolean value of color == “red”?

True

QUICK GAME: TRUE OR FALSE?

Fun time!!!

color = “blue”

what is the boolean value of color == “red”?

QUICK GAME: TRUE OR FALSE?

Fun time!!!

color = “blue”

what is the boolean value of color == “red”?

False

QUICK GAME: TRUE OR FALSE?

Fun time!!!

x = 5

y = 6

what is the boolean value of

x <= 5 and y > 5?

QUICK GAME: TRUE OR FALSE?

Fun time!!!

x = 5

y = 6

what is the boolean value of

x <= 5 and y > 5?

True and True = True

QUICK GAME: TRUE OR FALSE?

Fun time!!!

x = 5

y = 6

what is the boolean value of

x <= 5 and y > 10?

QUICK GAME: TRUE OR FALSE?

Fun time!!!

x = 5

y = 6

what is the boolean value of

x <= 5 and y > 10?

True and False = **False**

QUICK GAME: TRUE OR FALSE?

Fun time!!!

x = 5

y = 6

what is the boolean value of

not (x <= 5 and y > 10)?

QUICK GAME: TRUE OR FALSE?

Fun time!!!

x = 5

y = 6

what is the boolean value of

not (x <= 5 and y > 10)?

not (True and False) = not (False) = **True**

IF, ELIF, ELSE STATEMENT

Time for some comparison...

```
if the statement is true:
    Do this
else:
    Do that
```

IF, ELIF, ELSE STATEMENT

Time for some comparison

```
x = 4
if x < 5:
    print("x is smaller than 5")
elif x < 6:
    print("x is smaller than 6")
else:
    print("x is neither smaller than 5 nor 6")
```

IF, ELIF, ELSE STATEMENT

Time for some comparison

```
x = 5
if x < 5:
    print("x is smaller than 5")
elif x < 6:
    print("x is smaller than 6")
else:
    print("x is neither smaller than 5 nor 6")
```

IF, ELIF, ELSE STATEMENT

Time for some comparison

```
x = 7
if x < 5:
    print("x is smaller than 5")
elif x < 6:
    print("x is smaller than 6")
else:
    print("x is neither smaller than 5 nor 6")
```

QUICK GAME!

What is the answer to this?

```
ball_color = “blue”

if ball_color == “red”:
    print(“The color of the ball is red.”)
elif ball_color == “yellow”:
    print(“The color of the ball is yellow.”)
else:
    print(“The color of the ball is blue.”)
```

QUICK GAME!

What is the answer to this?

```
ball_color = "blue"
```

Define ball as blue

```
if ball_color == "red":
    print("The color of the ball is red.")
elif ball_color == "yellow":
    print("The color of the ball is yellow.")
else:
    print("The color of the ball is blue.")
```

Check: Is ball red? No.

Check: Is ball yellow? No.

Check: Is ball blue? Yes!

LOOPS

For Loops

While Loops

FOR LOOPS

Time for some iteration

```
for [a variable] in [something]:
    Do something
    Do something
    Do something
    ...
```

FOR LOOPS

Time for some iteration

```
for x in range(5):
    print(x)
```

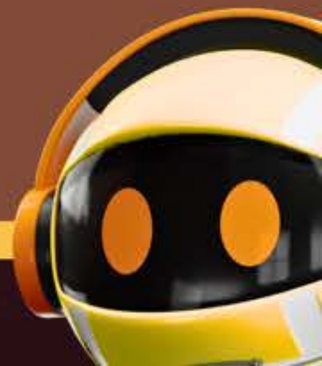
```
for number in range(5):
    print(number)
```

```
0
1
2
3
4
>
```

WHILE LOOPS

Time for some iteration

```
while True:
    Do something
    Do something
    Do something
    ...
```



WHILE LOOPS

Time for some iteration

```
x = 0


while x<5:
    print(x)
    x += 1
```

```
0
1
2
3
4
>
```

WHILE LOOPS

Time for some iteration

```
x = 0

while x<5:
    print(x, "<", 5, "is", (x<5))
    x += 1
```

```
0 < 5 is True
1 < 5 is True
2 < 5 is True
3 < 5 is True
4 < 5 is True
>
```

FUNCTIONS!!!

FUNCTIONS

A collection of code

```
def function_name():
    Do something
    Do something
    Do something...
```

FUNCTIONS

A collection of code

```
def my_function():
    print("Hello from a function")
```

FUNCTIONS

A collection of code

```
def my_function():
    print("Hello from a function")
```

FUNCTIONS

A collection of code

```
def my_function():
    print("Hello from a function")

my_function()
```

FUNCTIONS

A collection of code

```python
def my_function():
    print("Hello from a function")

my_function()
```

```
Hello from a function
>
```

FUNCTIONS

A collection of code

So, why is there a bracket beside the function?

FUNCTIONS

A collection of code

```
def function_name(parameters):
    Do something
    Do something
    Do something...
```

FUNCTIONS

A collection of code

```
def say_hello(name):
    print("Hello " + name + "!")
```

FUNCTIONS

A collection of code

```
def say_hello(name):
    print("Hello " + name + "!")

say_hello("katie")
say_hello("Bob")
```

→ ?

FUNCTIONS

A collection of code

```python
def say_hello(name):
    print("Hello " + name + "!")

say_hello("katie")
say_hello("Bob")
```

```
Hello katie!
Hello Bob!
>
```

FUNCTIONS

A collection of code

```
def say_hello(name, location):
    print("Hello " + name + " from " + location + "!")

say_hello("katie", "Mississauga")
say_hello("Bob", "Toronto")
```

FUNCTIONS

A collection of code

```
def say_hello(name, location):
    print("Hello " + name + " from " + location + "!")

say_hello("katie", "Mississauga")
say_hello("Bob", "Toronto")
```

```
Hello katie from Mississauga!
Hello Bob from Toronto!
>
```

FUNCTIONS

A collection of code

```
def say_hello(name, location):
    print("Hello " + name + " from " + location + "!")

say_hello("katie", "Mississauga")
say_hello(location="Toronto", name="Bob")
```

```
Hello katie from Mississauga!
Hello Bob from Toronto!
>
```

CHALLENGE!

1. **Story line or Character Journey Project!**
2. Take some input from user (name, age, certain preferences about topics, current day)
3. Calculate how many days are left to the weekend (if the day is a weekday)
4. Create if/else/elif statements to make decisions about certain inputs
5. Create a story template and implement the inputs into the story

THANK YOU!

Any other questions?