Warmup: World Pop

https://dmoj.ca/problem/p124ex4

Python
Strings

Before We Start

This is an **Intro to Python Lesson** for beginners, people with more experience are encouraged to join the **CP lessons** in **room 223**.

To get started: https://replit.com/repls.

For newcomers: Execs will help with setup.

Recap: Strings

A sequence of characters.

Enclosed in *double* or *single* quotes.

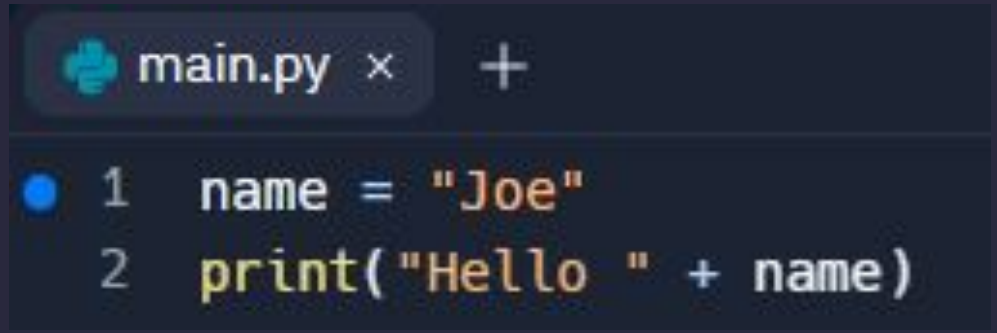
main.py

```
1  x = "I like dogs"
```

"Apple", 'Hello!', "I have 5 cats", 'Bye'

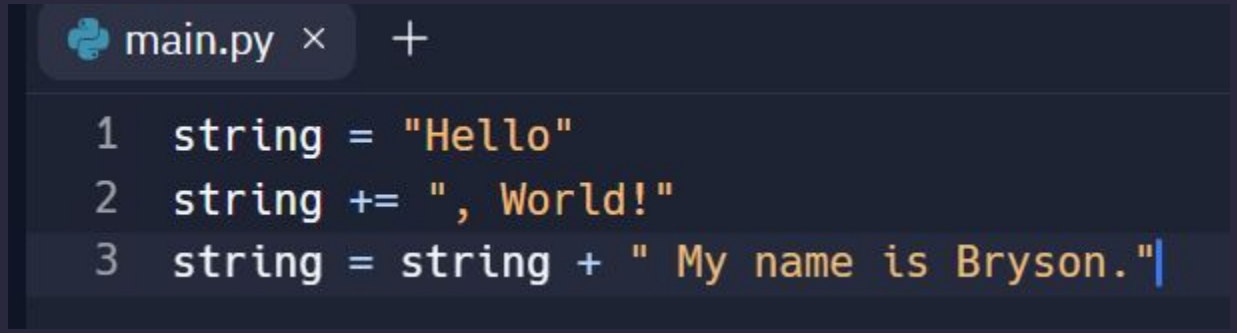
Recap: Combining Strings

We can combine strings together using the "**+**" symbol. This is known as **concatenation**.



Combining Strings

We can also add strings by using **+=** or **string = string + "hello"**



String Multiplication

Week 1:

```
1  print("Hello!")
2  print("Hello!")
3  print("Hello!")
4  print("Hello!")
5  print("Hello!")
6  print("Hello!")
7  print("Hello!")
8  print("Hello!")
9  print("Hello!")
10 print("Hello!")
```

Week 3:

```
15 for i in range(10):
16     print("Hello!")
```

Today:

```
19 print("Hello!\n"*10)
```

Character Escaping (Common ones)

`"\n"`

```
1 print("Hello, World!\nThis is a new line.")
```

```
Hello, World!
This is a new line.
```

`" \"Hi\" "`

```
1 print("\"I am the one who knocks!\" said Walter.")
```

```
"I am the one who knocks!" said Walter.
```

Full List

Quotes inside of Strings

" 'This is a quote' "

Enclose **single** quotation marks in a **double** quotation string

' "This is also a valid quote" '

Enclose **double** quotation marks in a **single** quotation string

String Slicing

```
greet = "Hello, World! I like pizza!"
```

greet[x]

Get a single character of the string at index x. Index counting starts at 0 (The first character has an index of zero, the second has an index of one…)

print(greet[4]) -> "o"

ALSO: **print(greet[-1]) -> "!"** **print(greet[-2]) -> "d"**

greet[x:y]

Get a part of the string: starting from the index x and ending with the index y (exclusive)

print(greet[5:9]) -> ", Wo"

String Slicing

```
greet = "Hello, World! I like pizza!"
```

greet[:x]
Get a part of the string: starting from the start and ending with the index x (exclusive)
print(greet[:6]) -> "Hello,"

greet[x:]
Get a part of the string: starting from the index x until the end of the string
print(greet[8:]) -> "orld!"

len()

len(string)
Gets the length of a string.

print(len("Hello"))
-> 5

print(len("Python"))
-> 6

The 'in' keyword

Use Case 1:

Check if a character or phrase exists in a string.

```
1 ▼ if "el" in "Hello, World!":
2       print("Yes")
3
4   print("hi" in "Hello,World!");
5
6   x = "Hello, World!"
7   y = "el"
8
9   print(y in x)
```

```
Yes
False
True
```

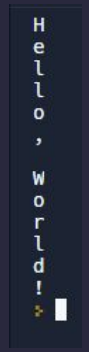
The 'in' keyword

Use Case 2:

String iteration (Loop through characters in a string)

```
1  string = "Hello, World!"
2  for i in range(len(string)):
3      print(string[i])
```

```
1  string = "Hello, World!"
2  for letter in string:
3      print(letter)
```



What's the difference?

<-

What's the Difference?

```
1  string = "Hello, World!"
2  for i in range(len(string)):
3      print(string[i])
```

Allows you to access the index of each character.

```
1  string = "Hello, World!"
2  for letter in string:
3      print(letter)
```

Iterates through each character in the string.

Cannot access the index.

String Methods

Full List

What's a method?

For now, a method is a piece of code someone else wrote that we can use on strings to do certain things on it.

More in-depth on methods in a *later meeting*.

String Methods

string.**capitalize()**

Make the first letter uppercase.

string.**startswith**(val)

Check if a string starts with a certain value

string.**count**(val)

Get the amount of times a certain value appears in a string

string.**endswith**(val)

Check if a string ends with a certain value

String Methods

string.index(val)

Get the index of a certain value in a string

string.isdigit()

Check if all characters in a string are digits

string.isalpha()

Check if all characters in a string are part of the alphabet

string.islower()

Check if all characters in a string are lower case.

String Methods

string.isupper()

Check if all characters in a string are uppercase.

string.upper()

Make all the characters in a string uppercase.

string.lower()

Make all the characters in a string lowercase.

string.replace(val1, val2)

Replace a certain value in a string with another value.

Python Practice

Vote Count (J2)

https://dmoj.ca/problem/ccc14j2

English or French? (S1)

https://dmoj.ca/problem/ccc11s1

Occupy Parking (J2)

https://dmoj.ca/problem/ccc18j2

More Practice...

Happy or Sad*

https://dmoj.ca/problem/ccc15j2

Ljesnjak

https://dmoj.ca/problem/coci08c5p1

Rotating* Letters

https://dmoj.ca/problem/ccc13j2

French Homework*

https://dmoj.ca/problem/dmopc14ce1p1

Cyclic Shifts*

https://dmoj.ca/problem/ccc20j4

D-Mails

https://dmoj.ca/problem/dmopc15c5p2