SPAI

An AI Singapore Student Chapter

ML Bootcamp

Day 0

Scan to mark attendance

Scan the QR code to mark your attendance

Attendance

Breaks and Q&A

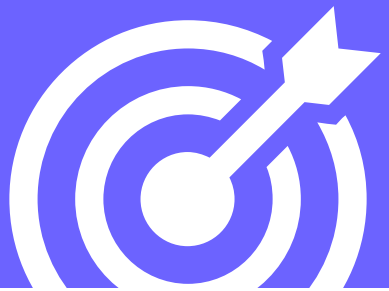
- [x] Several breaks every hour or so
- [x] Breaks will double as a Q&A session
- [x] Q&A session will also be present at the end of each day
- [x] If question isn’t urgent, please wait till the allocated time slots to ask your questions

Pre-requisites

You should have:

- [x] Read technical setup instructions for setting up necessary applications used in this bootcamp

- [x] Completed the Introduction to Python course on DataCamp



Learning Objectives

- [x] Understand what is python, and uses of notebook environments
- [x] Understand basic programming concepts
- [x] Carry out basic python programming using python

Note: Today's session is focused solely on recapping the concepts

Python & Notebook Environments

Python is a popular programming language with simple easy-to-understand syntax

What is Python?

What is Python?

It can be used to:

- [x] Create web applications and workflows
- [x] Connect to database systems
- [x] Handle big data
- [x] Rapid Prototyping

Notebook Environments

Colab

VS Code

Jupyter Lab

SPAI

Colab

Convenient to share with other people through Gmail

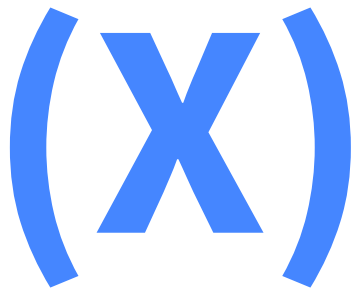
Can save files easily

Uses Google’s computing power instead of your device

Free of charge

SPAI

Variables (X)



Variables are a reference to a value contained in them

`What are Variables?`

SPAI

Variables

In mathematics, algebraic expressions contain a value

You can let various algebraic expressions contain different values by writing “let x = 4”

Same thing as variables in Python

(X) Variables Rules

A variable name…

- ☑ Must start with a letter or underscore
- ☑ Cannot start with a number
- ☑ Can only contain alpha-numeric characters and underscore
- ☑ Case-sensitive (age, Age and AGE are all different)

Allowed Variable Names

```
_John = 2
John = 4
A948 = 45

_John, John, A948
```

```
2, 4, 45
```

Not Allowed Variable Names

```
# Not Allowed
42RotiPrata = 42
```

```
  File "<ipython-input-2-c83853ff5faa>", line 2
    42RotiPrata = 42
      ^
SyntaxError: invalid syntax
```

```
%RotiPrata = 3
```

```
UsageError: Line magic function `%RotiPrata` not found.
```

SPAI

Case-Sensitive Variable Names

```
curry = 4
Curry = 5

curry, Curry
```

```
4, 5
```

Knowledge Check

Variables

SPAI

Knowledge Check

Which of the following variable name is accepted?

```
A. _HelloWorld
B. SPAIBes+
C. M4ng035
D. &3rs0n
```

SPAI

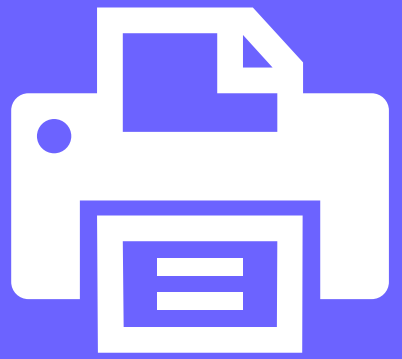
Knowledge Check

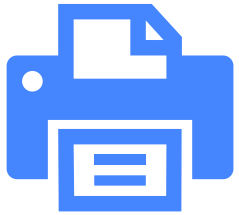
```
check1 = 4
check2 = 9

check1 + check2
```

```
>> What is the answer?
```

Printing & Comments

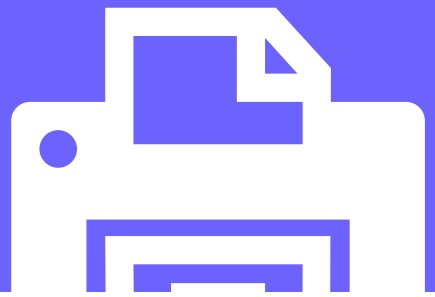




Printing what you want to see on your screen

What is Printing

SPAI



Ways to print

You can…

- [x] Type print(*“my text”*)

- [x] Declare a variable and print the value of the variable

Ways to print

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

```
Hello World!
```

```
wlc_msg = 'Welcome to SPAI Bootcamp!'

print(wlc_msg)
```

```
Welcome to SPAI Bootcamp!
```

SPAI

Comments allows you to annotate a piece of code. This allows you or others to understand the code.

What are comments

How to comment

You cannot comment in the middle of the code

```
print("Hello World!" #hello there)
```

```
  File "<ipython-input-6-57c74179e9b8>", line 1
    print("Hello World!" #hello there)
                                      ^
SyntaxError: unexpected EOF while parsing
```

Only at the end, top or bottom of the code

```
# Comment at the top
print('Hello World!') # At the end of the code
# Or at the bottom
```

```
Hello World
```

Knowledge Check

Printing & Commenting

SPAI

Knowledge Check

```
b = "Your laptop costs ${:.2f}"
price = 1999.8749

print(b.format(price))
```

```
>> What is the output?
A. Your laptop costs 1999.8749
B. Your laptop costs 1999.874
C. Your laptop costs 1999.88
D. Your laptop costs 1999.87
```

SPAI

Knowledge Check

A.

```
# Test print("Test Print")
```

```
Test print
```

B.

```
print("Hello There!" # No)
```

```
Hello There!
```

C.

```
print(#"Goodbye!")
```

```
Goodbye!
```

D.

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

```
#Hello World!
```

Practice Time!

5 Minutes

Please attempt Lab Exercise 1

We will go through the exercise later

SPAI

Time's up

We will now go through the exercises

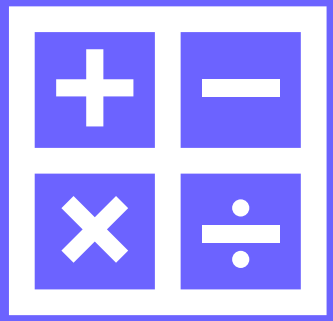
SPAI

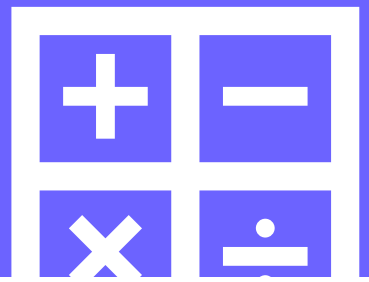
Break and Q&A

15 Minutes

SPAI

Python Operators





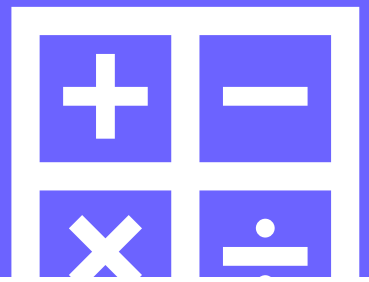
Arithmetic Operators

Math

- Addition: +
- Subtraction: —
- Multiplication: ×
- Division: ÷

Python

- Addition: `+`
- Subtraction: `-`
- Multiplication: `*`
- Division: `/`



Mod Operator

- [x] 23 mod 7

```
23%7
```

```
2
```

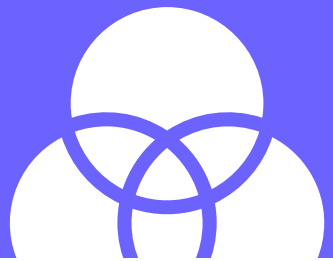
$$\begin{array}{r} 03 \\ 7\overline{)23} \\ -21 \\ \hline R2 \end{array}$$

- [x] 127 mod 8

```
127%8
```

```
7
```

$$\begin{array}{r} 015 \\ 8\overline{)127} \\ -08 \\ \hline 47 \\ -40 \\ \hline R7 \end{array}$$



Comparison Operator

>	<	>=	<=	==	!=
More than	Less than	More than or equal to	Less than or equal to	Logical Equal	Not Equal

SPAI

Knowledge Check

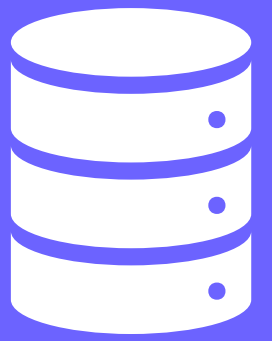
```
((4+5) * 2 > 18)
```

```
>> What is the answer?
A. True
B. 18
C. False
D. 14
```

SPAI

5.1

Data Types



7 Data Types

Data Types

- Strings
- Integers
- Float
- Boolean
- Lists
- Tuple
- Dictionary

Representation

- str
- int
- float
- bool
- list
- tuple
- dict

SPAI

Representation

```
"This is a string", "3" # Strings
1, 2, 3, 4 # Integers
2.4, 3.0, 1.9 # Floats
True, False # Boolean
# List, Tuples and Dictionaries will be covered later
```

SPAI

Finding out Data types

- Use “type” to get the data type of a variable

```
a = 4.0
type(a) # type(a) finds out data type of a variable
```

```
float
```

- Use the data type with parentheses to change the data type

```
a = int(a) # int(a) converts variables to integers
type(a)
```

```
float
```

Knowledge Check

```
type("hello"), type("3.0"), type(int(4.0)), type(2.9), type(False)
```

```
>> What is the answer?
A. str, float, float, float, bool
B. str, str, int, float, bool
C. str, int, int, float, bool
D. str, str, int, float, str
```

SPAI

Practice Time!

8 Minutes

Please attempt Lab Exercise 2

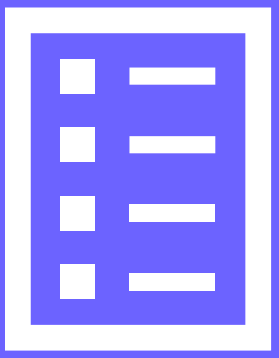
We will go through the exercise later

SPAI

5.2

Data Types

List



List

List can contain different data types

```
grades = ['DIST', 'A', 'B+', 'B', 'C+', 'C']

type(grades)
```

```
list
```

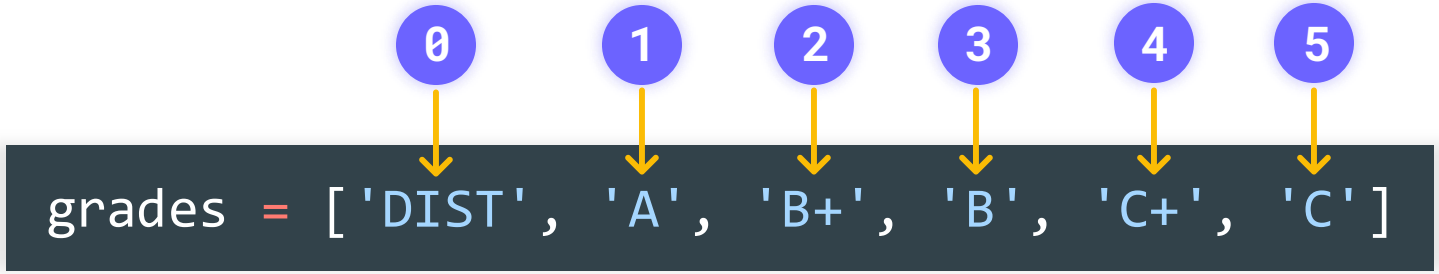
```
gpa = [4.0, 4.0, 3.5, 3.0, 2.5, 2.0]

type(gpa)
```

```
list
```

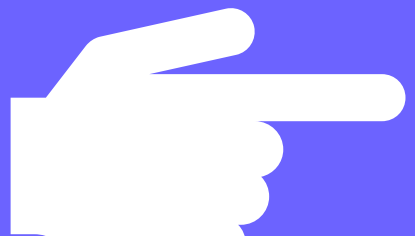
SPAI

List Indexing



All elements in a list have an assigned index **starting with 0**

SPAI



Index Selection

Select a specific element in the list using its index encapsulated with square brackets

```
grades = ['DIST', 'A', 'B+', 'B', 'C+', 'C']

print(f'The 2nd element of the list is: {grades[1]}')
print(f'The 4nd element of the list is: {grades[3]}')
```

```
The 2nd element of the list is: A
The 4th element of the list is: B
```

SPAI

Knowledge Check

```
fruits = ['apple', 'banana', 'kiwi', 'mango']
print(fruits[2])
```

```
>> What is the answer?
A. apple
B. banana
C. kiwi
D. mango
```

SPAI

5.3

Data Types

Tuples

Tuples function the same as lists.

They are however **immutable and unchangeable** while list are mutable

`What are tuples`

SPAI

Properties of tuple

- Tuples encloses its elements with parentheses => ()
- Tuples are immutable (cannot be changed)

Creating Tuples

Similar to how lists are declared, just replace square brackets with parentheses

```
tuple1 = ('A+', 'A', 'B+', 'B', 'C+', 'C')
tuple2 = (4.0, 4.0, 3.5, 3.0, 2.5, 2.0)

print(tuple1, tuple2)
print("Type of tuple1:", type(tuple1))
print("Type of tuple2:", type(tuple2))
```

```
('A+', 'A', 'B+', 'B', 'C+', 'C') (4.0, 4.0, 3.5, 3.0, 2.5, 2.0)
Type of tuple1: <class 'tuple’>
Type of tuple2: <class 'tuple'>
```

SPAI

Index Selection

Select a specific element in the tuple using its index encapsulated with square brackets

```
tuple1 = ('A+', 'A', 'B+', 'B', 'C+', 'C')
tuple2 = (4.0, 4.0, 3.5, 3.0, 2.5, 2.0)

print("First element of tuple1:", tuple1[0])
print("Last element of tuple2:", tuple2[-1])
print("Forth element of tuple2:", tuple2[3])
```

```
First element of tuple1: A+
Last element of tuple2: 2.0
Forth element of tuple2: 3.0
```

Knowledge Check

Which statement is correct?

A. Tuples are mutable

B. Lists and tuples have the exact same properties

C. Tuples encloses its elements with curly brackets, { }

D. Tuples can be accessed by using indexing.

5.4

Data Types

Dictionaries

Dictionaries are used to store data in key-value pairs

What are dictionaries

Creating Dictionaries

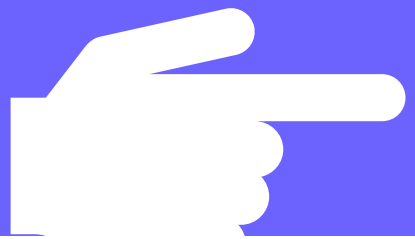
Enclose all key:value pairs with curly brackets

```
grades = {
    "DIST": 4.0, "A": 4.0, "B+": 3.5, "B": 3.0,
    "C+": 2.5, "C": 2.0, "D+": 1.5, "D": 1.0, "D-": 0.5
}

print(grades)
```

```
{'DIST': 4.0, 'A': 4.0, 'B+': 3.5, 'B': 3.0, 'C+': 2.5, 'C': 2.0, 'D+': 1.5, 'D': 1.0, 'D-': 0.5}
```

SPAI



Accessing Dictionaries

- Accessing using .get() function

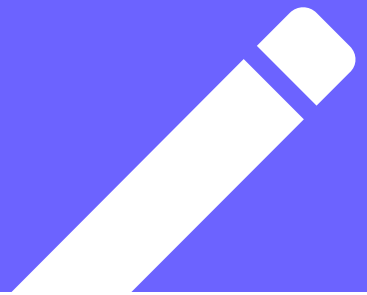
```
print(grades.get('A'))
```

```
4.0
```

- Using key encapsulated with square brackets

```
print(grades['A’])
```

```
4.0
```



Editing Dictionaries

Reassign a key's value

```
print("Before:\n", grades)
grades["D-"] = 0
print("After:\n", grades)
```

```
Before:
{'DIST': 4.0, 'A': 4.0, 'B+': 3.5, 'B': 3.0, 'C+': 2.5, 'C': 2.0, 'D+': 1.5, 'D':
1.0, 'D-': 0.5}
After: {'DIST': 4.0, 'A': 4.0, 'B+': 3.5, 'B': 3.0, 'C+': 2.5, 'C': 2.0, 'D+':
1.5, 'D': 1.0, 'D-': 0}
```

Adding Values

Set a new key to a value

```
print("Before:\n", grades)
grades["F"] = 0
print("After:\n", grades)
```

```
Before: {'DIST': 4.0, 'A': 4.0, 'B+': 3.5, 'B': 3.0, 'C+': 2.5, 'C': 2.0, 'D+':
1.5, 'D': 1.0, 'D-': 0}
After: {'DIST': 4.0, 'A': 4.0, 'B+': 3.5, 'B': 3.0, 'C+': 2.5, 'C': 2.0, 'D+':
1.5, 'D': 1.0, 'D-': 0, 'F': 0}
```

Knowledge Check

```
countries = {"Australia": "Canberra", "Canada": "Ottawa",
    "England": "London", "Japan": "Tokyo", "Malaysia": "Kuala Lumpur"}

print(countries["England"])
```

```
>> What is the answer?
A. Canberra
B. Ottawa
C. London
D. Tokyo
```

SPAI

Practice Time!

15 Minutes

Please attempt Lab Exercises 3 and 4

We will go through the exercises later

SPAI

Time's up

We will now go through the exercises

SPAI

Lunch Time

1 Hour

SPAI

6

Conditional Statements

Conditional statements

`if`

`elif`

`else`

Acts as a standalone statement

Required in all conditional statements

Does not execute if criteria(s) not satisfied

if

If statement

```
burger = 2.50
burgerset = 5

if burgerset > burger:
    print("Burger set is more expensive than the burger!")
else:
    print("Burger set is cheaper than the burger!")
```

```
Burger set is more expensive than the burger!
```

SPAI

Not compulsory

Stands for “else if”

Used for 2 or more conditions

elif

Elif statement

```
yourincome = 190000
if yourincome >= 320000:
    totaltax = 44550 + (yourincome - 320000) * 0.22
elif yourincome >= 280000:
    totaltax = 36550 + (yourincome - 280000) * 0.20
elif yourincome >= 240000:
    totaltax = 28750 + (yourincome - 240000) * 0.195
elif yourincome >= 200000:
    totaltax = 21150 + (yourincome - 200000) * 0.19
elif yourincome >= 160000:
    totaltax = 13950 + (yourincome - 160000) * 0.18
elif yourincome >= 120000:
    totaltax = 7950 + (yourincome - 120000) * 0.15
elif yourincome >= 80000:
    totaltax = 3350 + (yourincome - 80000) * 0.115
elif yourincome >= 40000:
    totaltax = 550 + (yourincome - 40000) * 0.07
elif yourincome >= 30000:
    totaltax = 200 + (yourincome - 30000) * 0.035
elif yourincome >= 20000:
    totaltax = (yourincome - 20000) * 0.02
else:
    totaltax = 0
print("You total chargeable income tax is $" + "{:.2f}".format(totaltax))
```

```
Your total chargeable income tax is $19350.00
```

SPAI

Not compulsory

Used at the end of all other compulsory statements

Used to execute an alternative scenario when “if” criteria not met

else

Else statement

```
yourincome = 1000
if yourincome >= 320000:
    totaltax = 44550 + (yourincome - 320000) * 0.22
elif yourincome >= 280000:
    totaltax = 36550 + (yourincome - 280000) * 0.20
elif yourincome >= 240000:
    totaltax = 28750 + (yourincome - 240000) * 0.195
elif yourincome >= 200000:
    totaltax = 21150 + (yourincome - 200000) * 0.19
elif yourincome >= 160000:
    totaltax = 13950 + (yourincome - 160000) * 0.18
elif yourincome >= 120000:
    totaltax = 7950 + (yourincome - 120000) * 0.15
elif yourincome >= 80000:
    totaltax = 3350 + (yourincome - 80000) * 0.115
elif yourincome >= 40000:
    totaltax = 550 + (yourincome - 40000) * 0.07
elif yourincome >= 30000:
    totaltax = 200 + (yourincome - 30000) * 0.035
elif yourincome >= 20000:
    totaltax = (yourincome - 20000) * 0.02
else:
    totaltax = 0
print("You total chargeable income tax is $" + "{:.2f}".format(totaltax))
```

```
Your total chargeable income tax is $0.00
```

SPAI

Knowledge Check

```
>> What is the answer?
A. $13950
B. $21150
C. $17550
D. $10950
```

```python
yourincome = 180000

if yourincome >= 320000:
    totaltax = 44550 + (yourincome - 320000) * 0.22
elif yourincome >= 280000:
    totaltax = 36550 + (yourincome - 280000) * 0.20
elif yourincome >= 240000:
    totaltax = 28750 + (yourincome - 240000) * 0.195
elif yourincome >= 200000:
    totaltax = 21150 + (yourincome - 200000) * 0.19
elif yourincome >= 160000:
    totaltax = 13950 + (yourincome - 160000) * 0.18
elif yourincome >= 120000:
    totaltax = 7950 + (yourincome - 120000) * 0.15
elif yourincome >= 80000:
    totaltax = 3350 + (yourincome - 80000) * 0.115
elif yourincome >= 40000:
    totaltax = 550 + (yourincome - 40000) * 0.07
elif yourincome >= 30000:
    totaltax = 200 + (yourincome - 30000) * 0.035
elif yourincome >= 20000:
    totaltax = (yourincome - 20000) * 0.02
else:
    totaltax = 0
print("You total chargeable income tax is $" + "{:.2f}".format(totaltax))
```

SPAI

Practice Time!

8 Minutes

Please attempt Lab Exercises 5

We will go through the exercises later

SPAI

Times up

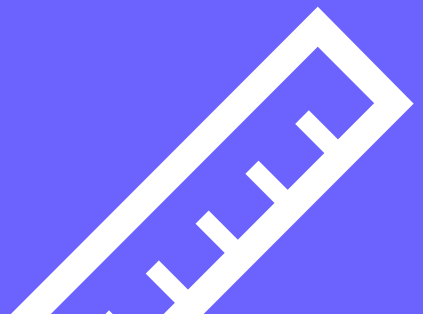
We will now go through Lab Exercise 5

SPAI

6.1

Loops

For Loop



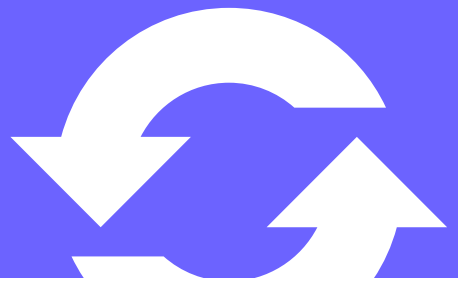
len()

```
gradeslist = ['DIST', 'A', 'B+', 'B', 'C+', 'C']
tuple2 = (4.0, 3.0, 3.5, 3.0, 2.5, 2.0)
gradesdict = {'DIST':4.0, 'A':4.0, 'B+':3.5, 'B':3.0, 'C+':2.5, 'C':2.0, 'D+':1.5
, 'D':1.0, 'D-':0.5}
test = "This is a string!"

print(f'This is the length of the list: {len(gradeslist)}')
print(f'This is the length of the tuple: {len(tuple2)}')
print(f'This is the length of the dictionary: {len(gradesdict)}')
print(f'This is the length of the string: {len(test)}')
```

```
This is the length of the list: 6
This is the length of the tuple: 6
This is the length of the dictionary: 9
This is the length of the string: 17
```

SPAI



For Loop

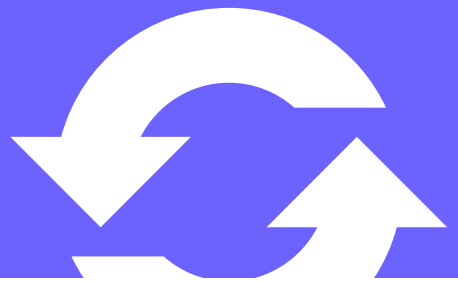
0 1 2

```
fruits = ['apple', 'banana', 'cherry']

for x in fruits:
    print(x)
```

```
apple
banana
cherry
```

SPAI



For Loop

0 1 2

```
fruits = ['apple', 'banana', 'cherry']

for x in range(len(fruits)):
    print(x, fruits[x])
```

```
1 apple
2 banana
3 cherry
```

Because “fruits” contains 3 elements, the loop repeats “for x in range(**3**)” times.

When using range, the number in the parentheses is a non-inclusive limit. Hence, x has the values 0, 1 and 2 and will stop at 2.

SPAI

Knowledge Check

```
for i in range(9):
    if i % 2 != 0:
        print("Hello")
```

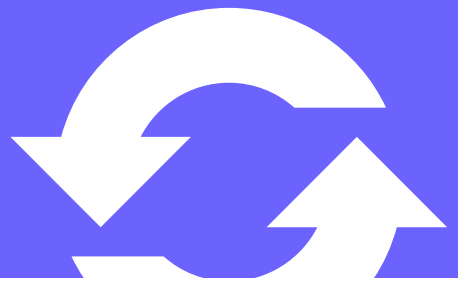
>> How many times will "Hello" be printed?

SPAI

6.2

Loops

While Loop



While Loop

While condition

```
counter = 4

while counter > 0: #condition
    print(counter)
    counter -= 1
```

```
4
3
2
1
```

SPAI

Knowledge Check

```
subjects = ['Math', 'English', 'Physics', 'Biology', 'Chemistry']

while len(subjects) - 1 > 0:
    print(subjects[0])
    subjects = subjects[1:]
```

>> At which round of the loop will “Chemistry” be printed?

SPAI

Practice Time!

20 Minutes

Please attempt Lab Exercises 6

We will go through the exercises later

SPAI

Time's up

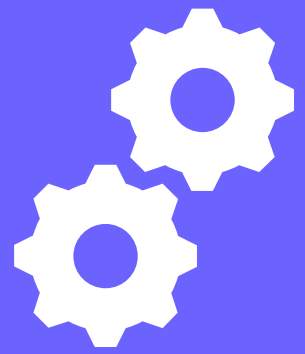
We will now go through Lab Exercise 6

Break and Q&A

15 Minutes

SPAI

Functions and Inputs



Defining Functions

```
def greetings(name):
    print(f'Hello {name}')

greetings('Tony')
```

```
Hello Tony
```

SPAI

Return Statements

```
def calculator(num1, num2, operand):
    operations = ['+', '-', '*', '/']
    for i in range(len(operations)):
        if operations[i] == operand:
            if operations[i] == '+':
                return(num1 + num2)
            elif operations[i] == '-':
                return(num1 - num2)
            elif operations[i] == '*':
                return(num1 * num2)
            elif operations[i] == '/':
                return(num1 / num2)

calculator(4, 3, '*')
```

```
12
```

```
def calculator(num1, num2, operand):
    operations = ['+', '-', '*', '/']
    for i in range(len(operations)):
        if operations[i] == operand:
            if operations[i] == '+':
                print(num1 + num2)
            elif operations[i] == '-':
                print(num1 - num2)
            elif operations[i] == '*':
                print(num1 * num2)
            elif operations[i] == '/':
                print(num1 / num2)

calculator(4, 3, '*')
```

```
12
```

SPAI

Function Scope

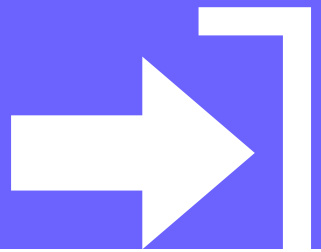
```
def functionscope():
    print("Inside the function")

print("Outside the function")
```

```
Outside the function
```

The code inside the function will not run called upon. The print statement for “Outside the function” will run, as it is outside the function.

SPAI



Input

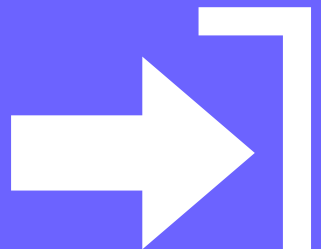
```
input("What is your name?")
```

```
What is your name?
```

```
input("What is your name?")
```

```
What is your name?Tony
'Tony'
```

SPAI



Input

```
answer = 'Yes'

while answer.lower() != 'no': # ".lower()" converts the whole string int
o lowercase letters.
    answer = input('Do you wish to continue? ')
```

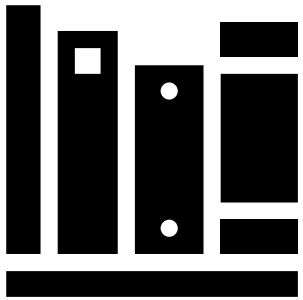
```
Do you wish to continue? Yes
Do you wish to continue? yEs
Do you wish to continue? yeS
Do you wish to continue? No
```

SPAI

Libraries

Libraries are a huge amount of pre-built code prepared by others to perform an action.

What are libraries



Libraries

Convenient for coders as code is ready out of the box

Simply import libraries into code

Saves time

Libraries

```
from math import log

print(log(4, 5))
```

```
0.8613531161467861
```

With one argument, it returns the natural logarithm of *x* (to base e).

With two arguments, it returns the logarithm of *x* to the given *base*, calculated as log(x)/log(base).

Practice Time!

10 Minutes

Please attempt Lab Exercises 7

We will go through the exercises later

SPAI

Time's up

We will now go through Lab Exercise 7

SPAI

Practice Time!

10 Minutes

Please attempt Lab Exercises 8

We will go through the exercises later

SPAI

Time's up

We will now go through Lab Exercise 8

SPAI

Break and Q&A

15 Minutes

SPAI

Practice Time!

8 Minutes

Please attempt Lab Exercises 9

We will go through the exercises later

SPAI

Time's up

We will now go through Lab Exercise 9

SPAI

Practice Time!

20 Minutes

Please attempt Lab Exercises 10

We will go through the exercises later

Time's up

We will now go through Lab Exercise 10

SPAI

Scan to mark attendance

Scan the QR code to check out

Check Out

SPAI