PYTHON WORKSHOP

Python Fundamentals

Prepared by the Artificial Intelligence Research Center (AIRC), Ajman University, Ajman, UAE

Workshop Outlines

- ~~Workshop Overview~~
- ~~Project Demo~~
- ~~Introduction to Python~~
- ~~Starting with Colab~~
- ~~Variables and simple data types~~
- ~~Comment and Print~~
- String manipulation
- If Statements
- Loops
- Lists
- Dictionaries
- Functions
- File Manipulation
- Final Project

Day 2 Outlines

- Data Types Review
- String Manipulation
- If Statements

1. Data Types Review

Review basic syntax and solve the exercises

Exercise 1

- Take the name, age and height of user and assign them to variables: name, age, height respectively
- Print the message :

```
Your name is :John
Your age is: 30
Your height is: 170
```

Start

Exercise 2

- Create a dictionary (name it data) ,using the information we obtained from previous exercise
- Print the dictionary as following:

```
{'age': '30', 'height': '170', 'name': 'John'}
```

Start

Exercise 3

- Create the following list of years as the following

Years = [2000, 2001, 2002, 2003,2004,2005]

- Add the years 2020, 2021, 2022 to the end of the list
- Print the full list, it should look like the following list :

```
[2000, 2001, 2002, 2003, 2004, 2005, 2020, 2021, 2022]
```

Start

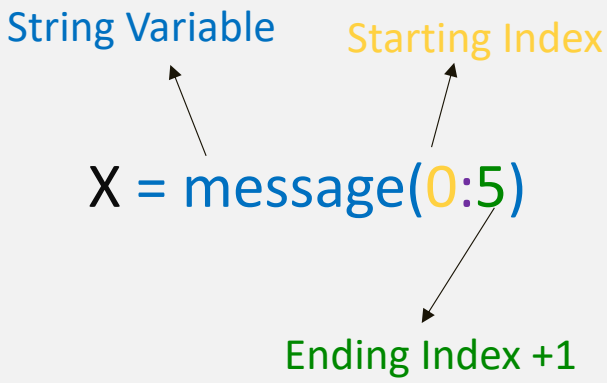
2. String Manipulation

The process of handling and analyzing strings. It involves various operations concerned with modification of strings

String Slicing

```
message  = "hello world"
hello_msg1= message[0:5]
hello_msg2 = message[:-6]
```

h	e	l	l	o		w	o	r	l	d
0	1	2	3	4	5	6	7	8	9	10
-11	-10	-9	-8	-7	-6	-5	-4	-3	-2	-1



python

String Case Conversion

Upper Case

```
message  = "hello world"
msg=message.upper()
print(msg)
```

HELLO WORLD

Lower Case

```
message  = "hello world"
msg=message.lower()
print(msg)
```

hello world

Title Case

```
message  = "hello world"
msg=message.title()
print(msg)
```

Hello World

String Count()

Return the count of occurrences of a keyword in a string

```
message  = "hello world"
count_x=message.count('x')
count_L=message.count('l')
print(count_x)
print(count_L)
```

```
0
3
```

String Variable

X = message.count('keyword')

Substring I want to count occurrences in the string

Note : the keyword parameter is case sensitive

python

String Find()

Return the starting index of a keyword in a string

```
message  = "hello world"
find_world=message.find('world')
find_x=message.find('x')
print(find_world)
print(find_x)
```

```
6
-1
```

String Variable

X = message.find('keyword')

Substring I want to search for in the string

Note : the keyword parameter if not found , -1 will be returned

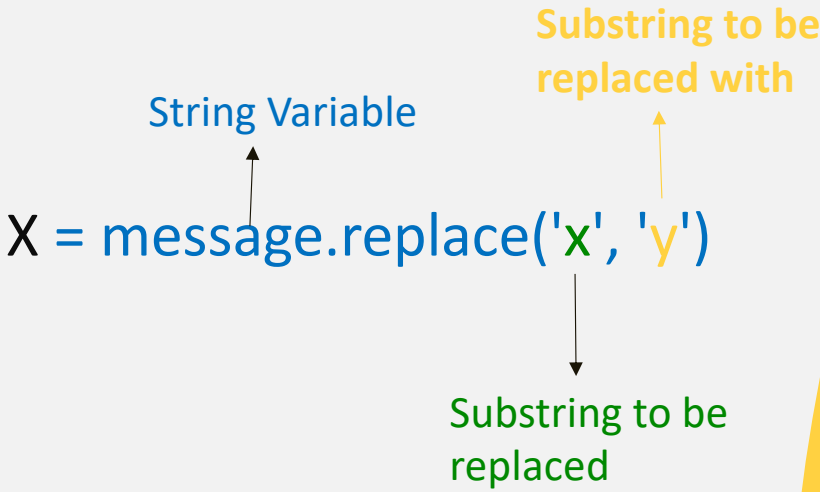
python

String Replace()

Replace every occurrence of keyword in a string with another keyword

```
message  = "hello world"
name= ", python students"
New_msg=message.replace(' world', name)
print(New_msg)
```

hello, python students



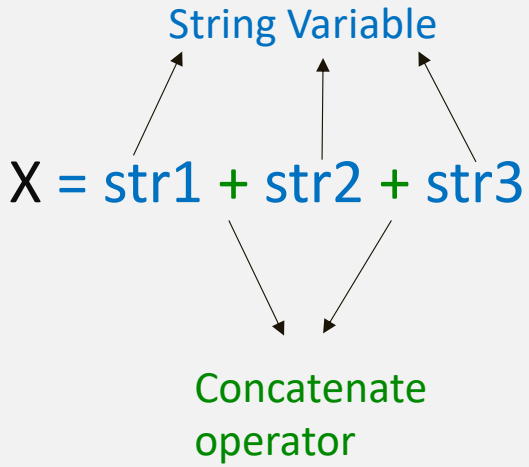
python

String Concatenate

The + operator concatenates strings. It returns a string consisting of the operands joined together

```
str1  = "hello "
str2  = " world "
str3  = ", again"
full_str= str1 + str2 + str3
print(full_str)
```

hello world, again



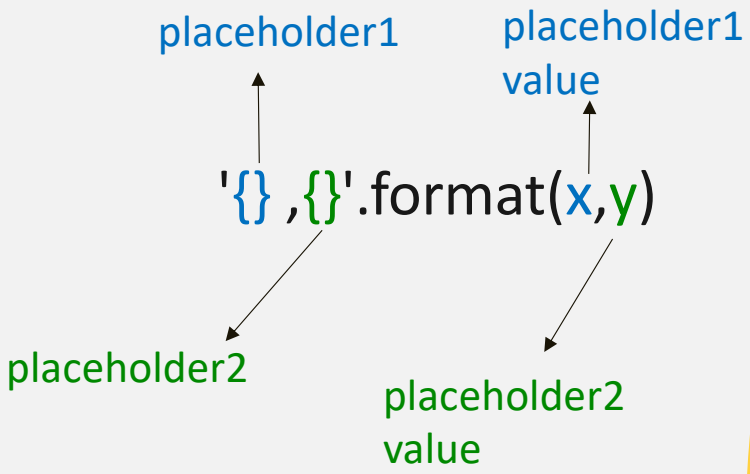
python

String Formatting

Method formats the specified value(s) and insert them inside the string's placeholder.

```
str1  = "hello"
str2  = " world"
intg  = 555
full_str= ''{}{}, this is {}'.format(str1,str2, intg)
print(full_str)
```

hello world, this is 555



python

4. Data Types and String Manipulation tutorial

Practice makes perfect

Exercise 4

- Create the variable String_exercise = "This is a tutorial session for string manipulation"
- Make "manipulation" upper case and title the rest of the string
- The final string should look like the following :

```
'This Is A Tutorial Session For String MANIPULATION'
```

Start

Exercise 5

- Replace "Is A" with "was a hopefully useful"
- The final string should look like the following :

```
'This was a hopefully usefull Tutorial Session For String MANIPULATION'
```

Start

5. If Statements

Direct the code according to conditions

If Statement – Basic Syntax

Main Keywords:

- If → if
- Else if → elif
- Else → else

```
if test expression :
  Body_Of_If
elif test expression :
  Body_Of_elif
else :
  Body_Of_else
```

Comparison operators

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

Logical operators

and	True if both sides are true	LeftSide and Rightside
or	True if at least one of the sides is true	LeftSide or Rightside
not	True if operand is false (complements the operand)	not (TheVariable/expression)

If Statement

Convert text to code :

If the number **is zero**, **then** print **"The number is zero"**. **Else if** the number is **greater than zero**, print **"The number is positive". Else** print **"The number is negative".**

```
if (num==0):
  print('The number is zero')
elif (num>0):
  print('The number is positive')
else :
  print('The number is negative')
```


python

Exercise

- Get two numbers x and y from the user .
- Convert the following conditions to “if-elif-else” code.

> if x equals y, then print “x is equal to y”. Else if x is greater than y, print “x is greater than y”. Else print “y is greater than x”.

Exercise

- Write a python program that takes a week number (1-7) as an input and print the corresponding day of week name using if- elif- else statement :
  - Such that 1 represents Monday, 2 Tuesday, 3 Wednesday, … , 7 Sunday.
  - In case the user input was not a number between 1 and 7 then print "Invalid Input! Please enter week number between 1-7.
- The Expected outputs

Case1

```
Please enter a number from 1 to 7 : 2
Tuesday
```

Case2

```
Please enter a number from 1 to 7 : 13
Invalid Input! Please enter week number between 1-7.
```