LAB #01: Introduction to Python

Lab Objective:

To introduce students with python programming language.

Lab Description:

Installation:

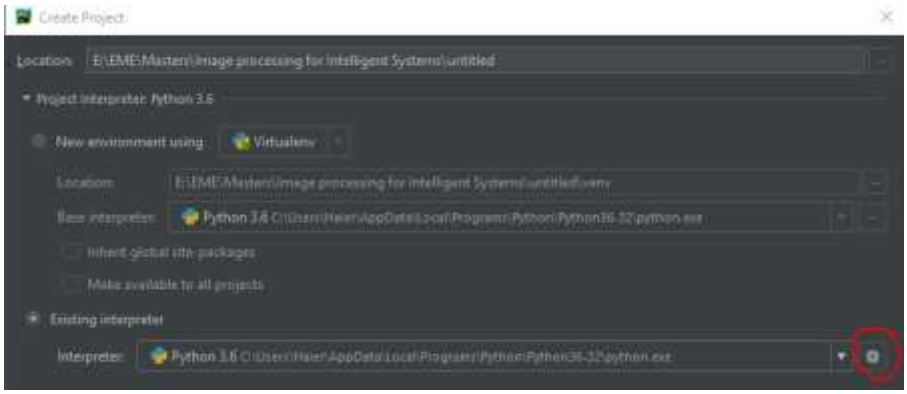
Download and install Python interpreter and PyCharm IDE from the following links below.

Python interpreter: https://www.python.org/downloads/

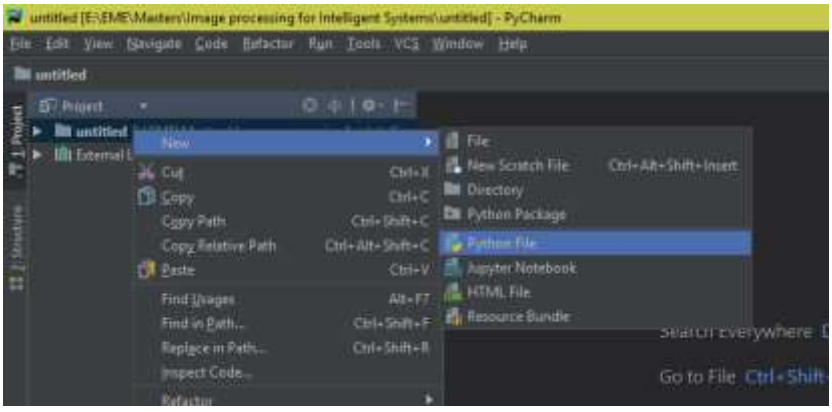
PyCharm IDE: https://www.jetbrains.com/pycharm/download/download-thanks.html?platform=windows&code=PCC

Setup:

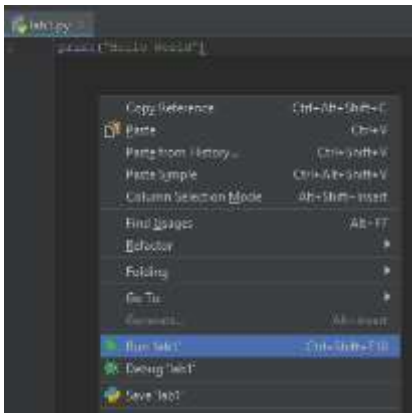
1. Create 'New Project'
2. Now Check 'Existing interpreter' and Add local python interpreter that you've installed, and click Create.



3. Now Right click on the folder and create new python file.



4. After writing your python code right click on the window and run the project.



Python is an interpreted high-level programming language for general-purpose programming. Python is meant to be an easily readable language. Its formatting is visually uncluttered, and it often uses English keywords where other languages use punctuation. Unlike many other languages, it does not use curly brackets to delimit blocks, and semicolons.

Operation	**Syntax**	**Explanation / Example**	**Fill the outputs of all the print statements**
Comment	#	# This is a comment.	
Print (is used to display anything in console window)	print ( )	print (10) print (12+6) print (5, end=" ") print ("END")	
Operators	+ plus - minus * multiply ** power / divide // divide and floor % modulus	print (5**2) print (5//2) print (5/2) print (5%2)	
Variables (Python automatically guess the data type. There is no need to explicitly define data type in python)	x = 5 y, z = 3.14, 17.2	right side is evaluated first and then assigned to left side with corresponding values print (x + y) print(type(x)) print(type(y))	
Strings	a="py" b='charm'	both single and double quotes works exactly same print (a + b) print (a, b) print (a*5) print ("hello' \"world\" ")	

Operation	**Syntax**	**Explanation / Example**	**Fill the outputs of all the print statements**
Lists List in memory stores references to objects. Each memory location is a pointer to an object. There is no obligation of similar data types	x=int (input (“Enter a number”) y = 3.14 z = "HELLO" li = [x, y, z, 4]	input ( ) function always input string, int ( ) function is used to convert string to int print(li)	
List Indexing # From left to right: 0 → 1 → 2 # From right to left: -1 → -2 → -3	Q = li [2] [-4]	print (Q)	
List Slicing list [start: end: step size] start is inclusive and end is exclusive defaults values are list [0 : end : 1]	R = li [1:3]	print (li [1:3]) print (li [0:4:2]) print (li [:]) print (li [0:]) print (li [:3]) print (li [2] [1:4])	
Copy The assignment copies the reference to the original list while slicing creates a new list	w = [1, 2, 3, 4] x = w y = w [:]	x [0] = 6 y [1] = 9 print(w)	
Indentation	x = 2 if x==10: print("inside”) print("inside") print("outside")	Whitespace (spaces and tabs) at the beginning of the logical line is used to determine the indentation level of the logical line, which in turn is used to determine the grouping of statements.	

Boolean operations	< (less than)
	> (greater than)
	<=(less than/equal to)
	>=(greaterthan/equal)
	== (equal to)
	! = (not equal to)
	not (Boolean NOT)
	and (Boolean AND)
	or (Boolean OR)
		if not x:
		if x==2 and y>4:
		print (2==4)	

Operation	**Syntax**	**Explanation / Example**	**Fill the outputs of all the print statements**

if
If the Boolean expression evaluates to true, then the if block of code will be executed
	number = 23
	if number == 24:
	print ('equal')
	elif number<24:
	print ('less')
	else:
	print ('greater')
		If statement does not include brackets and ends with colons. The if block is determine by indentation level	

while Repeats a statement or group of statements while a given condition is true	number = 23 while number<30: print(number) number+=1		

for
Execute a sequence of statements multiple times and abbreviates the code that manages the loop variable
	for i in [2,3,4,1,5]:
	print(i)
	for j in range(1,9,2):
	print(j)
		Range function is used to generate a list of numbers, which is generally used to iterate over with for loops	

Functions
A function is a block of code which only runs when it is called
	def max (x, y):
	if x > y:
	return x
	else:
	return y
	m=max (3,5)
	print(m)		

Lab Tasks:

1:

x = [[1, 2, 3, 4, 5], [21, 22, 23, 24, 25], [31, 32, 33, 34, 35]]

- Write python code using python indexing and slicing for the following output. Use only one print statement for each output.
  i. [21, 22, 23, 24, 25]
  ii. 3
  iii. [32, 33]
  iv. [1, 3, 5]
- Declare y = [0, 0, 0], now using for loop write average of first list in list 'x' on first index of list y and so on. The print(y) should give the following output
  - [3.0, 23.0, 33.0] // average of [1, 2, 3, 4, 5] = 3.0

- Declare z = [0, 0, 0, 0, 0], now using for loop write average of each index of each list in 'x' on corresponding index of list y. The print(z) should give the following output
    - [17.66, 18.66, 19.66, 20.66, 21.66] // average of [1, 21, 31] = 17.66

2:

x = [1, 3, 5, 6, 7, 8, 6, 1, 2, 3]

y = [0, 0, 0, 0, 0, 0, 0, 0]

- Write python code using while loop that write average of first three items on first index of y and so on. The print(y) should give the following output
    - [3.0, 4.666666666666667, 6.0, 7.0, 7.0, 5.0, 3.0, 2.0]
- Define a function that takes list as argument and returns the average of it. Then calculate the average of x and y.

Home Tasks:

Write a code that takes two integers from user as input and store in x and y variables. Now generate a list (w) of x elements and each element contains y random integers ranging from 1 to 10 and display this list.

- Now ask user to enter a number from 1 to 10 and display the number of times that integer occurs in the whole list (w).
- Write a code that calculate mean, median and mode of the list (w).
- Write a code that make another list (u) of same dimensions and compare each number of original list (w) with the average of it calculated in the previous part, if the number is less than the average, the code should place 0 on the corresponding index in newly created list (u) else it should place 10. Display the list (u).

THINK!!

1. While copying, why x changed w while y didn't? See above.
2. What will be the data type of result of a Boolean operation?
3. In Task 1, x is a list of what dimensions? Is it 3x5, 15 or just 3?
4. Can a list be created from range function?