Tech Module Exercises: 0 / 61

Python

* Mini Projects

|  |  |  |
| --- | --- | --- |
| S.No. | Mini-project Description | Topics Covered |
| 1 | Employee Management | Python |

* + In this TECH Module, you are expected to complete the below Mini-Project(s)
* Topics to Learn

To complete the above project, you will need to lean the below technical topics-

* + **Introduction to Python**

Learning Material for **Introduction to Python**

Below is the learning material that you are expected to read along with completion of the hands-on assignments. The material is mentioned is the order in which it should be read.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| No | Material Title | Material Location | Type of Material | Classification |
| 1. | Introduction to Python | Introduction to Python.pdf | PDF | Mandatory |
| 2. | Installing Python | <https://realpython.com/courses/installing-python-windows-macos-linux/> | Web | Suggestive |

* + - Hands-on Assignments for **Introduction to Python**
      * No Hands-on Assignments for this topic
  + **Language Basics**

Learning Material for **Language Basics**

Below is the learning material that you are expected to read along with completion of the hands-on assignments. The material is mentioned is the order in which it should be read.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| No | Material Title | Material Location | Type of Material | Classification |
| 1. | Language Basics | Language Basics.pdf | PDF | Mandatory |
| 2. | Operators | <https://realpython.com/python-or-operator/> | Web | Suggestive |
| 3. | Variables in Python | <https://realpython.com/courses/variables-python/> | Web | Suggestive |
| 4. | Data Types in Python | <https://realpython.com/courses/python-data-types/> | Web | Suggestive |
| 5. | Beginner Tips | <https://realpython.com/courses/python-beginner-tips/> | Web | Suggestive |

* + - Hands-on Assignments for **Language Basics**
      * No Hands-on Assignments for this topic
  + **Flow control statements**

Learning Material for **Flow control statements**

Below is the learning material that you are expected to read along with completion of the hands-on assignments. The material is mentioned is the order in which it should be read.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| No | Material Title | Material Location | Type of Material | Classification |
| 1. | Flow control statements | Flow control statements.pdf | PDF | Mandatory |
| 2. | Conditional statements | <https://realpython.com/courses/python-conditional-statements/> | Web | Suggestive |
| 3. | Loops | <https://realpython.com/courses/how-to-write-pythonic-loops/> | Web | Suggestive |

* + - Hands-on Assignments for **Flow control statements**

Complete the below hands-on assignments before proceeding with the next Topic

|  |  |  |  |
| --- | --- | --- | --- |
| No. | Hands-on Assignment | Topic Covered | Status |
| 1. | Write a program to check if a given number is Positive, Negative, or Zero. | If else |  |
| 2. | Write a program to check if a given number is odd or even. | If else |  |
| 3. | Given tow non-negative vales, print true if last digit in both the numbers are the same, such as with 27 and 57.  lastDigit(7, 17) -> true  lastDigit(6, 17) -> false  lastDigit(3, 113) -> true | If else |  |
| 4. | Write a program to print numbers from 1 to 10 in a single row with on tab space. | for |  |
| 5. | Write a program to print even numbers between 23 and 57. Output should be printed in a sperate row. | for |  |
| 6. | Write a program to check if a given number is prime or not. | for |  |
| 7. | Write a program to print prime numbers between 10 and 99. | for |  |
| 8. | Write a program to print the sum of the digits of a given number.  Example:  I/P: 1234  O/P: 10 | while |  |
| 9. | Write a program to print the reverse a given number.  Example: 1  I/P: 1004  O/P: 4001 | while |  |
| 10. | Write a program to find if the given number is palindrome or not  Example: 1  I/P: 110011  O/P: 110011 is a palindrome.  Example: 2  I/P: 1234  O/P: 1234 is not a palindrome. | while |  |

* + **List**

Learning Material for **List**

Below is the learning material that you are expected to read along with completion of the hands-on assignments. The material is mentioned is the order in which it should be read.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| No | Material Title | Material Location | Type of Material | Classification |
| 1. | List | List.pdf | PDF | Mandatory |

* + - Hands-on Assignments for **List**

Complete the below hands-on assignments before proceeding with the next Topic

|  |  |  |  |
| --- | --- | --- | --- |
| No. | Hands-on Assignment | Topic Covered | Status |
| 1. | Write a program to create a list named MyList1 containing 5 integer items and display the list items. Access individual elements through index.( eg. Access 3rd and 5th element and display it)  MyList1 contains : 10,20,10,40,50 | List |  |
| 2. | Write a program to append a new item to the end of the list.  Eg: Append integer 60 to the MyList1 | List |  |
| 3. | Write a program to reverse the order of the items in the MyList1. | List |  |
| 4. | Write a program to print the number of occurrences of a specified element in a MyList1. | List |  |
| 5. | Create another list named MyList2 containing 2 integer items.  MyList2 contains : 11,21  Write a program to append the items of MyList2 to MyList1 in the front. | List |  |
| 6. | Write a program to insert a new item before the second element in the existing list MyList2 . | List |  |
| 7. | Write a program to remove the item from a specified index in the list MyList1. | List |  |
| 8. | Write a program to remove the first occurrence of a specified element from the list MyList1.  eg: 10 is repeated in MyList1 . Remove the first occurrence of it. | List |  |

* + **Dictionary**

Learning Material for **Dictionary**

Below is the learning material that you are expected to read along with completion of the hands-on assignments. The material is mentioned is the order in which it should be read.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| No | Material Title | Material Location | Type of Material | Classification |
| 1. | Dictionary | Dictionary.pdf | PDF | Mandatory |
| 2. | Dictionary | <https://realpython.com/courses/dictionaries-python/> | Web | Suggestive |

* + - Hands-on Assignments for **Dictionary**

Complete the below hands-on assignments before proceeding with the next Topic

|  |  |  |  |
| --- | --- | --- | --- |
| No. | Hands-on Assignment | Topic Covered | Status |
| 1. | Write a program to add a key and value pair to a dictionary named Mydic1.  Sample Dictionary : {0: 10, 1: 20}  Expected Result : {0: 10, 1: 20, 2: 30} | Dictionary |  |
| 2. | Write a program to concatenate the following dictionaries to create a new one. Sample Dictionary :  dic1= {1:10, 2:20}  dic2={3:30, 4:40}  dic3={5:50,6:60}  Expected Result : {1: 10, 2: 20, 3: 30, 4: 40, 5: 50, 6: 60} | Dictionary |  |
| 3. | Write a program to check if a given key already exists in a dictionary Mydic1. | Dictionary |  |
| 4. | Write a program to iterate over a dictionary Mydict1 using for loop and print the keys alone, values alone and both keys and values. | Dictionary |  |
| 5. | Write a program to prepare a dictionary Mydict2 where the keys are numbers between 1 and 15 (both included) and the values are square of the keys. | Dictionary |  |
| 6. | Write a program to sum all the values in a dictionary, considering the values will be of int type. | Dictionary |  |

* + **Tuple**

Learning Material for **Tuple**

Below is the learning material that you are expected to read along with completion of the hands-on assignments. The material is mentioned is the order in which it should be read.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| No | Material Title | Material Location | Type of Material | Classification |
| 1. | Tuple | Tuple.pdf | PDF | Mandatory |

* + - Hands-on Assignments for **Tuple**

Complete the below hands-on assignments before proceeding with the next Topic

|  |  |  |  |
| --- | --- | --- | --- |
| No. | Hands-on Assignment | Topic Covered | Status |
| 1. | Write a program to print the 4th element from first and 4th element from last in a tuple. | Tuple |  |
| 2. | Write a program to check whether an element exists in a tuple or not. | Tuple |  |
| 3. | Write a program to convert a list into a tuple. | Tuple |  |
| 4. | Write a program to find the index of an item in a tuple. | Tuple |  |
| 5. | Write a program to replace last value of tuples in a list to 100. Sample list: [(10, 20, 40), (40, 50, 60), (70, 80, 90)] Expected Output: [(10, 20, 100), (40, 50, 100), (70, 80, 100)] | Tuple |  |

* + **Set**

Learning Material for **Set**

Below is the learning material that you are expected to read along with completion of the hands-on assignments. The material is mentioned is the order in which it should be read.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| No | Material Title | Material Location | Type of Material | Classification |
| 1. | Set | Set.pdf | PDF | Mandatory |
| 2. | Set | <https://realpython.com/courses/sets-python/> | Web | Suggestive |

* + - Hands-on Assignments for **Set**

Complete the below hands-on assignments before proceeding with the next Topic

|  |  |  |  |
| --- | --- | --- | --- |
| No. | Hands-on Assignment | Topic Covered | Status |
| 1. | Write a program to remove a given item from the set. | Set |  |
| 2. | Write a program to create an intersection of sets. | Set |  |
| 3. | Write a program to create an union of sets. | Set |  |
| 4. | Write a program to find the maximum and minimum value in a set. | Set |  |

* + **String**

Learning Material for **String**

Below is the learning material that you are expected to read along with completion of the hands-on assignments. The material is mentioned is the order in which it should be read.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| No | Material Title | Material Location | Type of Material | Classification |
| 1. | String | String.pdf | PDF | Mandatory |

* + - Hands-on Assignments for **String**

Complete the below hands-on assignments before proceeding with the next Topic

|  |  |  |  |
| --- | --- | --- | --- |
| No. | Hands-on Assignment | Topic Covered | Status |
| 1. | Write a program to count the number of upper and lower case letters in a String. | String |  |
| 2. | Write a program that will check whether a given String is Palindrome or not. | String |  |
| 3. | Given an input string, return a new string which will display n copies of the first 2 characters of the original string where n is the length of the string. The input string length should be >=2. If input is “Wipro” then output should be “WiWiWiWiWi”. | String |  |
| 4. | Given an input string, if the first or last character of the string contains ‘x’, return a new string after removing the character ‘x’, else return the string unchanged.  input : “xHix”  output : “Hi”  input : “welcome”  output : “welcome” | String |  |
| 5. | Given as input a string and an integer n, return a new string made of n repetitions of the last n characters of the string. You may assume that n is between 0 and the length of the string inclusive.  For example : if the inputs given are “Wipro” and 3, then the output should be “propropro”. | String |  |

* + **Functions**

Learning Material for **Functions**

Below is the learning material that you are expected to read along with completion of the hands-on assignments. The material is mentioned is the order in which it should be read.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| No | Material Title | Material Location | Type of Material | Classification |
| 1. | Functions | Functions | PDF | Mandatory |
| 2. | Functions | <https://realpython.com/defining-your-own-python-function/> | Web | Suggestive |

* + - Hands-on Assignments for **Functions**

Complete the below hands-on assignments before proceeding with the next Topic

|  |  |  |  |
| --- | --- | --- | --- |
| No. | Hands-on Assignment | Topic Covered | Status |
| 1. | Write a function to return the sum of all numbers in a list.  Sample List : (8, 2, 3, 0, 7)  Expected Output : 20 | Functions |  |
| 2. | Write a function to return the reverse of a string.  Sample String : “1234abcd”  Expected Output : “dcba4321” | Functions |  |
| 3. | Write a function to calculate and return the factorial of a given number (a non-negative integer). | Functions |  |
| 4. | Write a function that accepts a string and prints the number of upper case letters and lower case letters in it. | Functions |  |
| 5. | Write a function to print the even numbers from a given list. List is passed as an argument to the function .  Sample List : [1, 2, 3, 4, 5, 6, 7, 8, 9]  Expected Result : [2, 4, 6, 8] | Functions |  |
| 6. | Write a function that takes a number as a parameter and checks whether the number is prime or not. | Functions |  |

* + **Modules**

Learning Material for **Modules**

Below is the learning material that you are expected to read along with completion of the hands-on assignments. The material is mentioned is the order in which it should be read.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| No | Material Title | Material Location | Type of Material | Classification |
| 1. | Modules | Modules.pdf | PDF | Mandatory |

* + - Hands-on Assignments for **Modules**

Complete the below hands-on assignments before proceeding with the next Topic

|  |  |  |  |
| --- | --- | --- | --- |
| No. | Hands-on Assignment | Topic Covered | Status |
| 1. | Create a file as math\_module.py and define the functions in it:  add(int,int)  sub(int,int)  mul(int,int)  div(int,int)  Import the module math\_module.py in another program called calculator.py file and use the functions defined in it. The program should be menu driven application working as a basic calculator. | Modules |  |
| 2. | Write a program that will import a datetime module and display today’s date | Modules |  |
| 3. | Write a program that will import a math module and print the sqrt of a given number. | Modules |  |

* + **Packages**

Learning Material for **Packages**

Below is the learning material that you are expected to read along with completion of the hands-on assignments. The material is mentioned is the order in which it should be read.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| No | Material Title | Material Location | Type of Material | Classification |
| 1. | Packages | Packages.pdf | PDF | Mandatory |

* + - Hands-on Assignments for **Packages**

Complete the below hands-on assignments before proceeding with the next Topic

|  |  |  |  |
| --- | --- | --- | --- |
| No. | Hands-on Assignment | Topic Covered | Status |
| 1. | Use the above created file math\_module.py as a package by moving it into new folder and call the functions in the package by creating another program . | Packages |  |

* + **Command Line Arguments**

Learning Material for **Command Line Arguments**

Below is the learning material that you are expected to read along with completion of the hands-on assignments. The material is mentioned is the order in which it should be read.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| No | Material Title | Material Location | Type of Material | Classification |
| 1. | Command Line Arguments | Command Line Arguments.pdf | PDF | Mandatory |
| 2. | Command Line Arguments | <https://realpython.com/python-command-line-arguments/> | Web | Suggestive |
| 3. | Command Line Arguments | <https://www.geeksforgeeks.org/command-line-arguments-in-python/> | Web | Suggestive |

* + - Hands-on Assignments for **Command Line Arguments**

Complete the below hands-on assignments before proceeding with the next Topic

|  |  |  |  |
| --- | --- | --- | --- |
| No. | Hands-on Assignment | Topic Covered | Status |
| 1. | Write a program to accept two numbers as command line arguments and display their sum. | Command Line Arguments |  |
| 2. | Write a program to accept a welcome message through command line arguments and display the file name along with the welcome message. | Command Line Arguments |  |
| 3. | Write a program to accept 10 numbers through command line arguments and calculate the sum of prime numbers among them. | Command Line Arguments |  |

* + **IO Operations**

Learning Material for **IO Operations**

Below is the learning material that you are expected to read along with completion of the hands-on assignments. The material is mentioned is the order in which it should be read.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| No | Material Title | Material Location | Type of Material | Classification |
| 1. | IO Operations | IO Operations.pdf | PDF | Mandatory |
| 2. | Input Output | <https://realpython.com/python-input-output/> | Web | Suggestive |
| 3. | Working with python | <https://realpython.com/working-with-files-in-python/> | Web | Suggestive |
| 4. | Print Function | <https://realpython.com/courses/python-print/> | Web | Suggestive |

* + - Hands-on Assignments for **IO Operations**

Complete the below hands-on assignments before proceeding with the next Topic

|  |  |  |  |
| --- | --- | --- | --- |
| No. | Hands-on Assignment | Topic Covered | Status |
| 1. | Write a program to read the entire content from a text file and display it to the user. | File Handling |  |
| 2. | Write a program to read first n lines from a text file. Get the value of n as input from the user. | File Handling |  |
| 3. | Write a program to accept input from user and append it to a text file. | File Handling |  |
| 4. | Write a program to read contents from a text file line by line and store each line into a list. | File Handling |  |
| 5. | Write a program to find the longest word from the txt file contents, assuming that the file will have only one longest word in it. | File Handling |  |
| 6. | Write a program to count the frequency of a user entered word in a text file. | File Handling |  |

* + **Exception Handling**

Learning Material for **Exception Handling**

Below is the learning material that you are expected to read along with completion of the hands-on assignments. The material is mentioned is the order in which it should be read.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| No | Material Title | Material Location | Type of Material | Classification |
| 1. | Exception Handling | Exception Handling.pdf | PDF | Mandatory |
| 2. | Exception Handling | <https://realpython.com/courses/introduction-python-exceptions/> | Web | Suggestive |

* + - Hands-on Assignments for **Exception Handling**

Complete the below hands-on assignments before proceeding with the next Topic

|  |  |  |  |
| --- | --- | --- | --- |
| No. | Hands-on Assignment | Topic Covered | Status |
| 1. | Write a program to accept two numbers from the user and perform division. If any exception occurs, print an error message or else print the result. | Exception Handling |  |
| 2. | Write a program to accept a number from the user and check whether it’s prime or not. If user enters anything other than number, handle the exception and print an error message. | Exception Handling |  |
| 3. | Write a program to accept the file name to be opened from the user, if file exists print the contents of the file in title case or else handle the exception and print an error message. | Exception Handling |  |
| 4. | Declare a list with 10 integers and ask the user to enter an index. Check whether the number in that index is positive or negative number. If any invalid index is entered, handle the exception and print an error message. | Exception Handling |  |

* Learning Outcomes

Upon completion of this TECH Module, you should be able to:

* + Understand Python architecture.
  + Understand how to install python.
  + Understand about Anaconda distribution.
  + Understand how to use Spyder IDE.
  + Understand How to write and run basic python scripts.
  + Understand about variables, operators, keywords.
  + Understand How to work with if, else, elif.
  + Understand How to work with for, while.
  + Understand How to work with range function.
  + Understand How to work with break, continue keywords.
  + Understand how to work with List.
  + Understand how to work with Dictionary.
  + Understand how to work with Tuple.
  + Understand how to work with Set.
  + Understand how to work with Strings.
  + Understand how to work with functions.
  + Understand about built-in functions.
  + Understand how to work with user defined functions.
  + Understand how to work with functions with arguments.
  + Understand about command line arguments.
  + Understand about sys module and argv variable.
  + Understand how to pass cmd line arguments from terminal.
  + Understand how to pass cmd line arguments from Spyder.
  + Understand how to access cmd line arguments.
  + Understand how to accept user inputs in your program.
  + Understand how to format the output.
  + Understand how to work with open function.
  + Understand how to read form a file, write to a file.
  + Understand how to create and delete files and folders.
  + Understand what is exception.
  + Understand what is exception handling.
  + Understand about built-in exceptions.
  + Understand how to use try and except blocks.
  + Understand how to use else and finally blocks.
* Sample Questions

After gaining knowledge of the above module, below are the possible interview questions that you should be able to confidently answer.

|  |  |
| --- | --- |
| No. | Questions |
| 1 | List out few scripting languages. |
| 2 | What is another name for Python 3.0? |
| 3 | Which command enables the interactive terminal mode? |
| 4 | List out few mutable data types. |
| 5 | Which function returns the location of an object? |
| 6 | Which operator does integer division? |
| 7 | What is type casting? |
| 8 | How to do multi line comment? |
| 9 | Which built in function returns a sequence of integers? |
| 10 | Can we use else with for? |
| 11 | Can we write statements after break keyword? |
| 12 | Which statement does nothing? |
| 13 | Does list maintain insertion order? |
| 14 | Can er access list items with negative index? |
| 15 | Which function empties the list? |
| 16 | Which class represents the group of characters? |
| 17 | Which is slicing? |
| 18 | Which function toggles the cases? |
| 19 | Does set take duplicate elements? |
| 20 | List few built in function. |
| 21 | Which keyword marks the start of the function? |
| 22 | Which keyword end the execution of a function? |
| 23 | Which allows you to logically organize your python files? |
| 24 | How to import a module with alias name? |
| 25 | Which command lists the built-in modules? |
| 26 | Can we import multiple modules with single import statement? |
| 27 | Which module is required to access cmd line arguments? |
| 28 | How to open a file in create mode? |
| 29 | Which function reads on line at a time? |
| 30 | Which module is required to delete files or folders? |
| 31 | What will be the output for print(5-‘A’)? |
| 32 | Which block is always executed irrespective of the exception? |
| 33 | Which block is executed when there is no exception? |
| 34 | Which is the base class for all the exception classes? |
| 35 | Do we need any module to implement exception handling? |