# System Requirements Specification Index

For

# Python Skills Evaluation

Version 1.0

IIHT Pvt. Ltd.
fullstack@iiht.com

# USE CASE DESCRIPTION

## Use Case 1

**Write a program to take input of string from user and count the number of uppercase alphabets, lowercase alphabets, digits and special characters. Store these counts in an array of size 4 in following order: Lowercase count, Uppercase count, Digit count, special char count.**

**Example:**

**Input:**Test,String12!

**Output:** [8, 2, 2, 2]

**Description**

1. Take console input as string, in **main ()** method, from user.
2. Pass the string to method **count_characters()** and write the logic in that method to count each character.
3. If input type is not string format, then raise TypeError from **count_characters()**function.
4. We have considered array stores count of char in sequence of lowercase alphabets, uppercase alphabets, digitsand special characters.
5. Return the array (list)having number oflowercase alphabets, uppercase alphabets, digits and special characters from **count_characters**() to main() and display the count.

## Use Case 2

**Write a program to take input of date from user in string format and display the day on which it falls. The code must fulfil following requirements:**

a. **Function for validating that the integer-representing month is between 1 & 12 and checking that the day part of the date object is within the correct range for a month.**
b. **Function for obtaining the week day from a given date.**

**Example:**
**Input:** 08-09-2021
**Output:** Wednesday

**Description**

1. Take console input of a Date as "DD-MM-YYYY" in string formatin **main ()** method from user.
2. Pass the date to method **get_week_day()**and write the logic in that method to find weekday on given date.
3. Define a method **get_month_days() ,** invoke this method from**get_week_day()**method by passing month and year.
4. Write the logic in **get_month_days()** method to find whether given month contains valid month from 1-12. If valid month, return number of days respected to given month. If monthis not in range return -1 from **get_month_days()** tomethod **get_week_day() .**
5. If input date is not string format, then raise TypeError from **get_week_day()** function.
6. Finally return the week day as String from **get_week_day()**to main(), as "MONDAY" or "TUESDAY" etc.

**Execution Steps to Follow:**

1. All actions like build, compile, running application, running test cases will be through Command Terminal.
2. To open the command terminal the test takers, need to go to

   Application menu (Three horizontal lines at left top) -> Terminal ->New Terminal.

3. The editor Auto Saves the code.
4. If you want to exit(logout) and to continue the coding later anytime (using Save & Exit option on Assessment Landing Page) then you need to use CTRL+Shift+B command compulsorily on code IDE. This will push or save the updated contents in the internal git/repository. Else the code will not be available in the next login.
5. These are time bound assessments the timer would stop if you logout and while logging in back using the same credentials the timer would resume from the same time it was stopped from the previous logout.
6. To run application for use case1 use the following command

   python3 count_chars.py

7. To run application for use case2 use the following command

   python3 dateandday.py

8. Mandatory: Before final submission run the following command

   python3 -m unittest

9. Once you are done with development and ready with submission, you may navigate to the previous tab and submit the workspace. It is mandatory to click on "Submit Assessment" after you are done with code.
10. You need to use CTRL+Shift+B - command compulsorily on code IDE, before final submission as well. This will push or save the updated contents in the internal git/repository, and will be used to evaluate the code quality.

-------*-------