System Requirements

Specification Index

For

Random Data Analysis using NumPy

(Topic: NumPy Arrays)

Version 1.0

Random Data Analysis Console

System Requirements Specification

Project Abstract

The Random Data Analysis Console is a Python application that generates a dataset of random integers and provides basic statistical analysis tools. The application calculates the mean and median of the dataset. This tool is useful for performing quick statistical analyses on random data, which can be applied in various scenarios such as simulations, data testing, or even educational purposes. The application aims to provide an easy-to-use interface for basic statistical computations on random datasets.

Business Requirements:

1. **Data Generation:**
   - **The system must generate a set of 100 random integers.**
   - **The range of integers must be between 1 and 1000.**
2. **Statistical Calculations:**
   - **The system must calculate the mean of the dataset.**
   - **The system must calculate the median of the dataset.**
   - **Ensure that proper error handling is implemented for edge cases (e.g., empty dataset).**

Constraints:

1. **Data Constraints:**
   - **The dataset must always contain exactly 100 random integers.**
   - **The integers must be within the range of 1 to 1000.**
2. **Output Constraints:**
   - **The mean and median must be calculated accurately and returned as floating-point values.**
   - **If the dataset is empty, the system must raise an error message.**
3. **Calculation Constraints:**
   - **The mean must be calculated using the standard arithmetic formula for the mean.**
   - **The median must be computed by sorting the dataset and selecting the middle value.**

Input Requirements:

1. **Data Generation:**
   - **No user input is required for generating the data.**
   - **The data will be automatically generated upon system initialization.**

Output Requirements:

1. **Display Format:**
    - **The mean and median must be displayed clearly after each calculation.**
    - **Proper error messages should be displayed if an issue arises (e.g., empty dataset).**

Required Output Format:

- **After generating the random dataset, the program should display the mean and median values in the following format:**
    - **"Mean: {value}"**
    - **"Median: {value}"**

Template Code Structure:

1. **Initialization:**
    - **Generate 100 random integers.**
    - **Store the data in an internal variable.**
2. **Statistical Calculations:**
    - **Calculate Mean: Compute the mean of the dataset.**
    - **Calculate Median: Compute the median of the dataset.**
3. **Error Handling:**
    - **Ensure proper handling of empty datasets or other edge cases.**
4. **Display Results:**
    - **Output the calculated mean and median in the required format.**

Execution Steps to Follow:

- All actions like build, compile, running application, running test cases will be through Command Terminal.
- To open the command terminal the test takers, need to go to Application menu (Three horizontal lines at left top) -> Terminal -> New Terminal
- This editor Auto Saves the code
- If you want to exit(logout) and 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.
- 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.
- To setup environment:

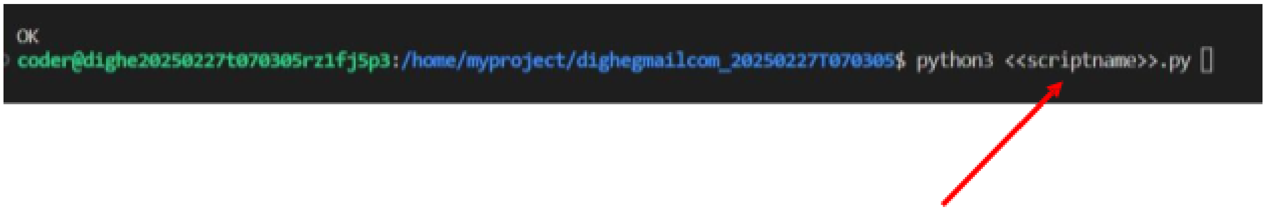
  You can run the application without importing any packages
- To launch application:
  python3 mainclass.py
- To run Test cases:

  python3 -m unittest

- Before Final Submission also, 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.

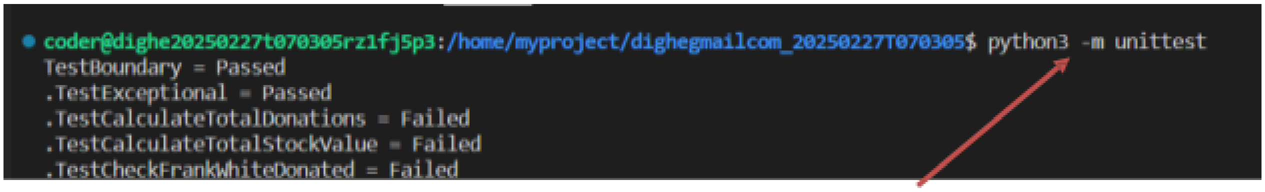
Screen shot to run the program

To run the application



```
OK
coder@dighe20250227t070305rz1fj5p3:/home/myproject/dighegmailcom_20250227T070305$ python3 <<scriptname>>.py
```

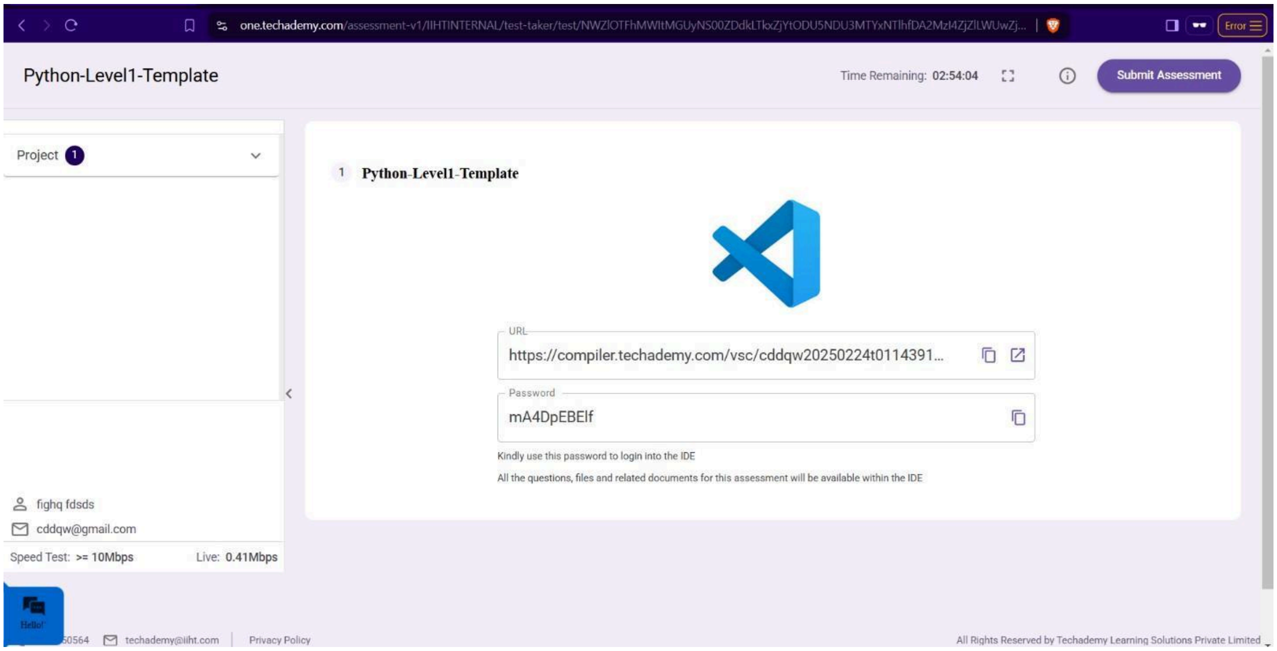
python3 mainclass.py



```
● coder@dighe20250227t070305rz1fj5p3:/home/myproject/dighegmailcom_20250227T070305$ python3 -m unittest
  TestBoundary = Passed
  .TestExceptional = Passed
  .TestCalculateTotalDonations = Failed
  .TestCalculateTotalStockValue = Failed
  .TestCheckFrankWhiteDonated = Failed
```

To run the testcase

python3 -m unittest



one.techademy.com/assessment-v1/IIHTINTERNAL/test-taker/test/NWZlOTFhMWItMGUyNS00ZDdkLTkxZjYtODU5NDU3MTYxNTlhfDA2MzI4ZjZlLWUwZj...

Python-Level1-Template

Time Remaining: 02:54:04

1 Python-Level1-Template

URL: https://compiler.techademy.com/vsc/cddqw20250224t0114391...

Password: mA4DpEBElf

Kindly use this password to login into the IDE

All the questions, files and related documents for this assessment will be available within the IDE

fighq fdsds

cddqw@gmail.com

Speed Test: >= 10Mbps Live: 0.41Mbps

50564 techademy@iiht.com | Privacy Policy

All Rights Reserved by Techademy Learning Solutions Private Limited

- **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.**