
System Requirements Specification Index

For

Python Basics and NumPy, Pandas

Usecase 2

1.0

Use Case1: Inventory Management (inventory_management.py)

The dataset provided to you

Dataset: Inventory (Item, Quantity) and Prices

```
inventory_data = [  
    ("Laptop", 5),  
    ("Mouse", 20),  
    ("Keyboard", 12),  
    ("Monitor", 3),  
    ("USB Drive", 50)  
]
```

```
item_prices = {  
    "Laptop": 1000,  
    "Mouse": 25,  
    "Keyboard": 80,  
    "Monitor": 300,  
    "USB Drive": 15  
}
```

1) Write a Python program to check and return low-stock items.

- Define a function `get_low_stock_items(inventory, reorder_level)`.
- The function should:
 - Loop through the inventory dataset.
 - Identify items with stock below the reorder level.
 - Return a list of such low-stock items.

2) Write a Python program to calculate the total inventory value.

- Define a function `calculate_inventory_value(inventory, prices)`.
- The function should:
 - Multiply each item's stock by its price.
 - Sum up the total inventory value.
 - Return the total inventory value.

Use Case:2 Weather Analysis (weather_analysis.py)

Dataset temperatures = {32.5, 34.0, 31.2, 29.8, 35.5}

1) Write a Python program to analyze weather data.

- Define a function `weather_analysis(temperatures)`.
- The function should:
 - Identify the highest and lowest temperatures.
 - Detect extreme temperatures (above 34°C or below 30°C).
 - Generate and return a formatted weather analysis report.

2) Write a Python program to save the weather analysis report to a file.

- Define a function `save_report_to_file(report, filename)`.
- The function should: `weather_analysis()`
 - Append the generated weather report to a text file.
 - Display a confirmation message when the report is saved.
 - Use a sample dataset of temperature readings in Celsius.

- Detect the Highest Temperature and low temperature
- Save the generated report using `save_report_to_file()`
- The text file you need to use is `weather_report.txt`

Sample format provided

Weather Analysis Report

Temperatures (°C): [temp1, temp2, temp3, temp4, temp5]

Highest Temperature: highest_temp

Lowest Temperature: lowest_temp

Extreme Temperatures Detected: [extreme_temp1, extreme_temp2]

Use Case3: Customer Purchase Analysis (`customer_purchase_analysis.py`)

Customer purchase dataset provided

```
data = {
    'Customer': ['Alice', 'Bob', 'Charlie', 'Diana', 'Eve'],
    'Items_Bought': [3, 7, 5, 2, 9],
    'Amount_Spent': [150.0, 400.0, 275.0, 100.0, 600.0]
}
```

1) Write a Python program to create and display a customer purchase dataset using Pandas.

- Define a function `create_dataframe()`.
- The function should:
 - Create a Pandas DataFrame containing customer purchase details.
 - Include the following columns: Customer, Items_Bought, Amount_Spent.
 - Return the DataFrame.

2) Write a Python program to calculate the average amount spent by customers.

- Define a function `calculate_average_spending(df)`.
- The function should:
 - Compute the average amount spent across all customers.
 - Return the calculated average.

3) Write a Python program to identify high-spending customers.

- Define a function `get_top_spenders(df, average_spending)`.
- The function should:
 - Filter and return a DataFrame of customers whose spending is above the average.

4) Write a Python program to analyze and display the customer purchase data.

- Implement a `main()` function that:
 - Calls `create_dataframe()` to generate the dataset.
 - Calls `calculate_average_spending(df)` to compute the average spending.
 - Calls `get_top_spenders(df, average_spending)` to identify high-spending customers.
 - Prints the customer purchase dataset and high-spending customers.

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. This editor Auto Saves the code
4. 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.
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 setup environment:

You can run the application without importing any packages

7. To launch application:

python3 customer_purchase_analysis.py

python3 inventory_management.py.py

python3 weather_analysis.py

To run Test cases:

python3 -m unittest

8. 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

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

To run the application

`python3 customer_purchase_analysis.py`

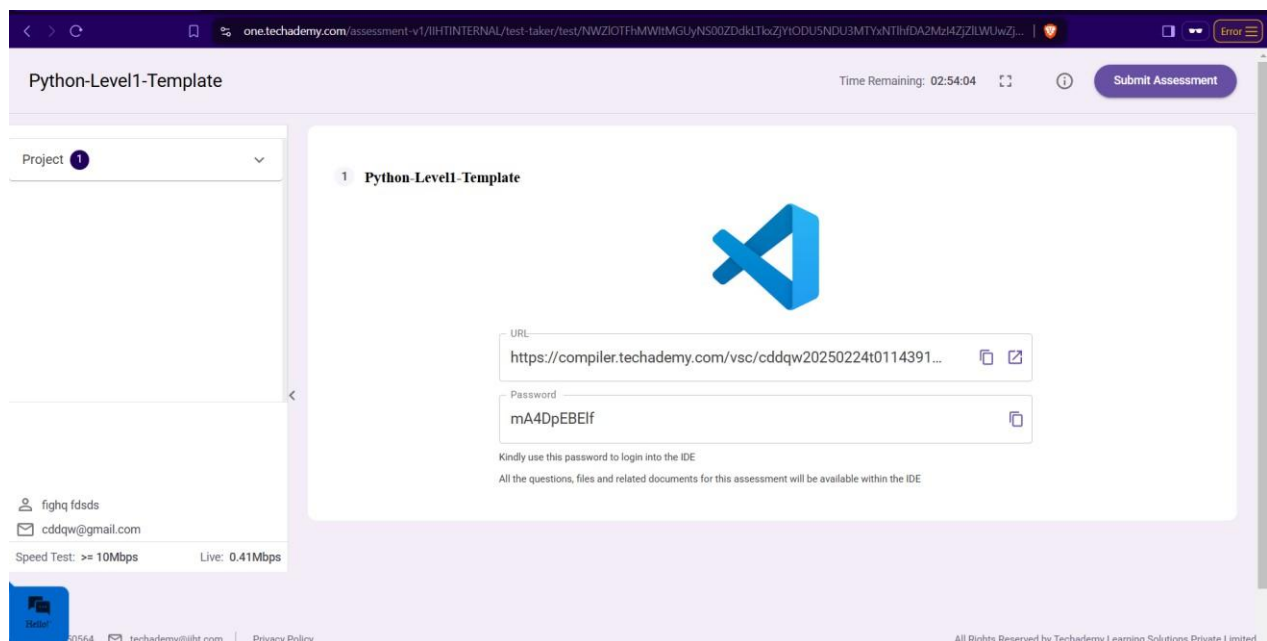
`python3 inventory_management.py.py`

`python3 weather_analysis.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`



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.