assessment instrument

**APPLY PRINCIPLES OF CREATING COMPUTER SOFTWARE BY DEVELOPING A COMPLETE PROGRAMME TO MEET GIVEN BUSINESS SPECIFICATIONS**

**US ID:** 115392

**NQF LEVEL:** 5

**CREDITS:** 12

**NOTIONAL HOURS: 1**20

Charles Chamusi

ASSESSMENT PROCESS FLOW

Assessment Plan agreed by candidate & completed by the assessor before the actual assessment

Knowledge Questionnaire conducted as per the Assessment Plan

Observation conducted as per the Assessment Plan

Portfolio of Evidence compiled as per the Assessment Plan

A detailed Assessor Report compiled & forwarded for Moderation

Record of Learning Updated

Appeal form completed by the candidate in the event of dispute

Feedback Report Completed by Assessor & individual feedback given to the candidate

Assessment Results Moderated

Action Plan Completed by Assessor

All records & evidence filed

Completed Assessor Report / Moderator Report / Record of Learning

Approval & Certification obtained

Certificate of Competencies issued to successful candidates

Register candidates on the Learner Record Database

Portfolio of Evidence submitted as per the Assessment Plan

**SETA**

**ASSESSMENT**

**CAND**

**I**

**DATE**

AGREED ASSESSMENT PLAN

|  |  |  |  |
| --- | --- | --- | --- |
| **Candidate's Name:** | **Unathi chonco** | | |
| **Assessor's Name:** | **Nicky Masiya** | | |
| **Standard Title:** | **APPLY PRINCIPLES OF CREATING COMPUTER SOFTWARE BY DEVELOPING A COMPLETE PROGRAMME TO MEET GIVEN BUSINESS SPECIFICATIONS** | | |
| **EVENT** | **DATE, TIME AND LOCATION** | **RESOURCES REQUIRED** | **EVIDENCE TO BE GENERATED** |
| Attend Training | 21-22/08/2024 | Training material, equipment as specified | Attendance Register |
| Complete formative assessment | 23/08/2024 | Formative workbook | Completed formative workbook |
| Complete summative assessment | 26/08/2024 | Summative workbook | Completed knowledge questionnaire |

Signature of Candidate: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ Signature of Assessor: \_\_\_\_\_\_\_\_

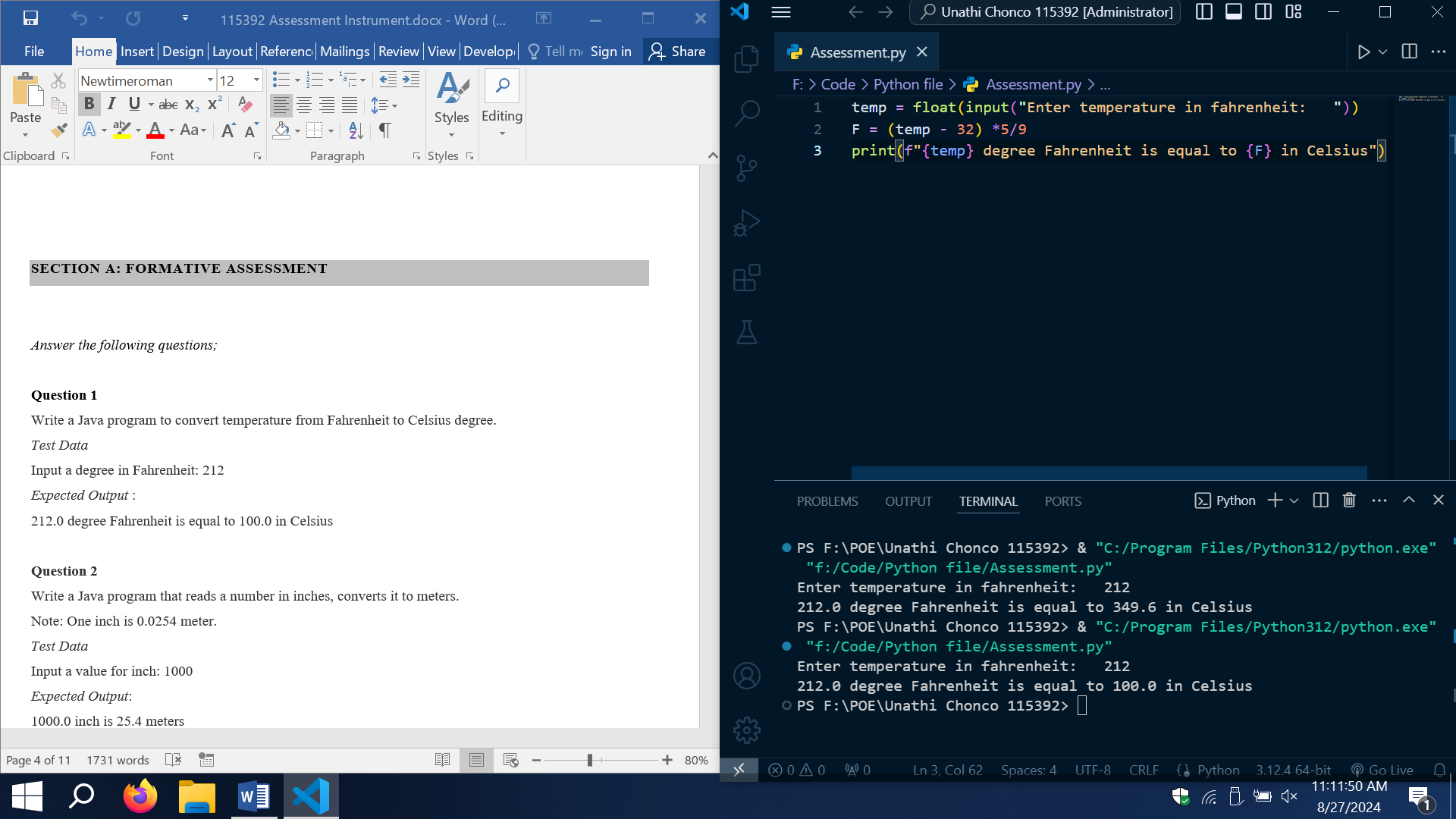
Date: \_\_21-22/08/2024\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

SECTION A: FORMATIVE ASSESSMENT

*Answer the following questions;*

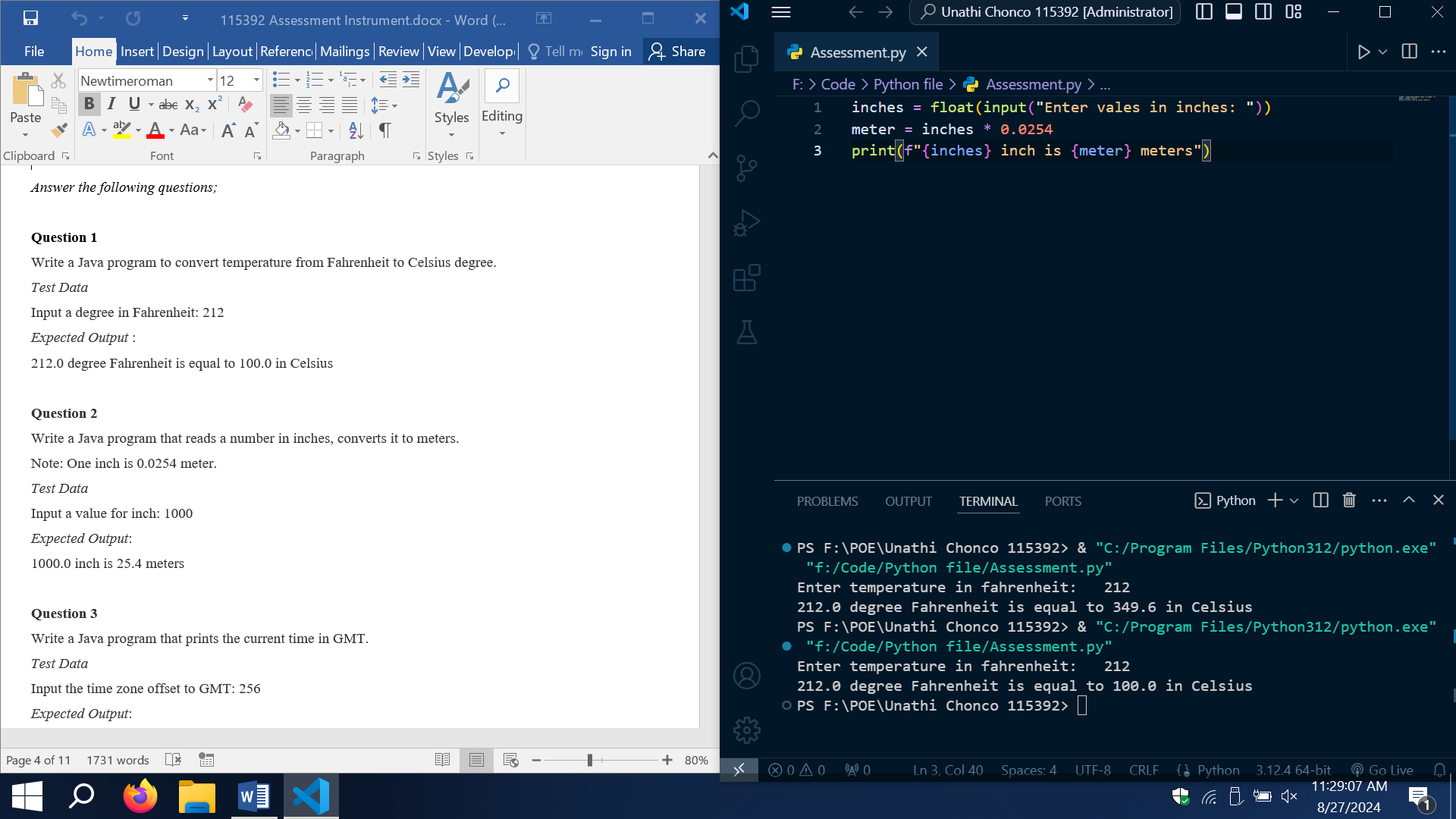
**Question 1**

Write a Java program to convert temperature from Fahrenheit to Celsius degree.    
Test Data  
Input a degree in Fahrenheit: 212  
Expected Output :  
212.0 degree Fahrenheit is equal to 100.0 in Celsius



**Question 2**

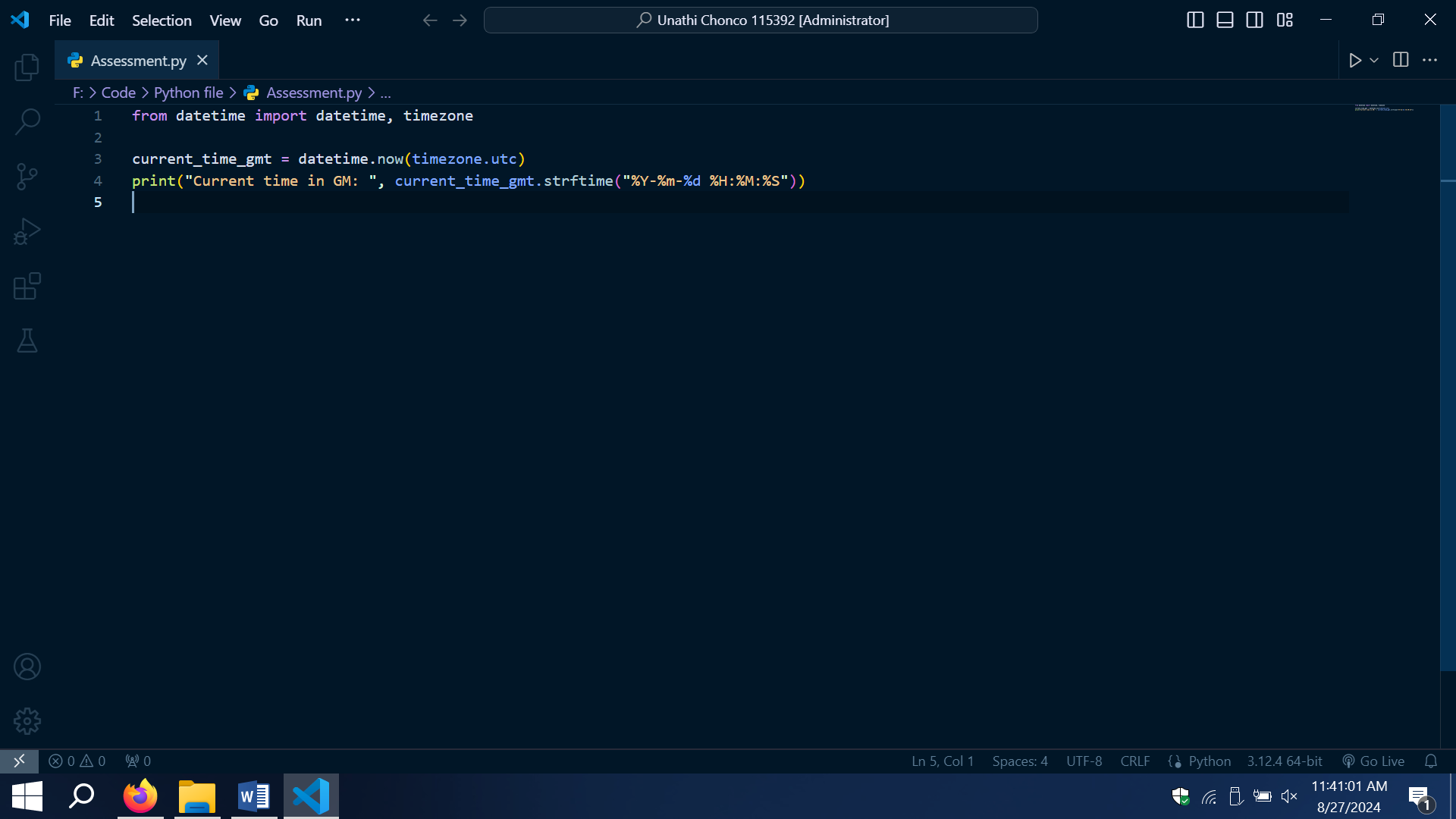
Write a Java program that reads a number in inches, converts it to meters.    
Note: One inch is 0.0254 meter.  
Test Data  
Input a value for inch: 1000  
Expected Output:   
1000.0 inch is 25.4 meters



**Question 3**

Write a Java program that prints the current time in GMT.

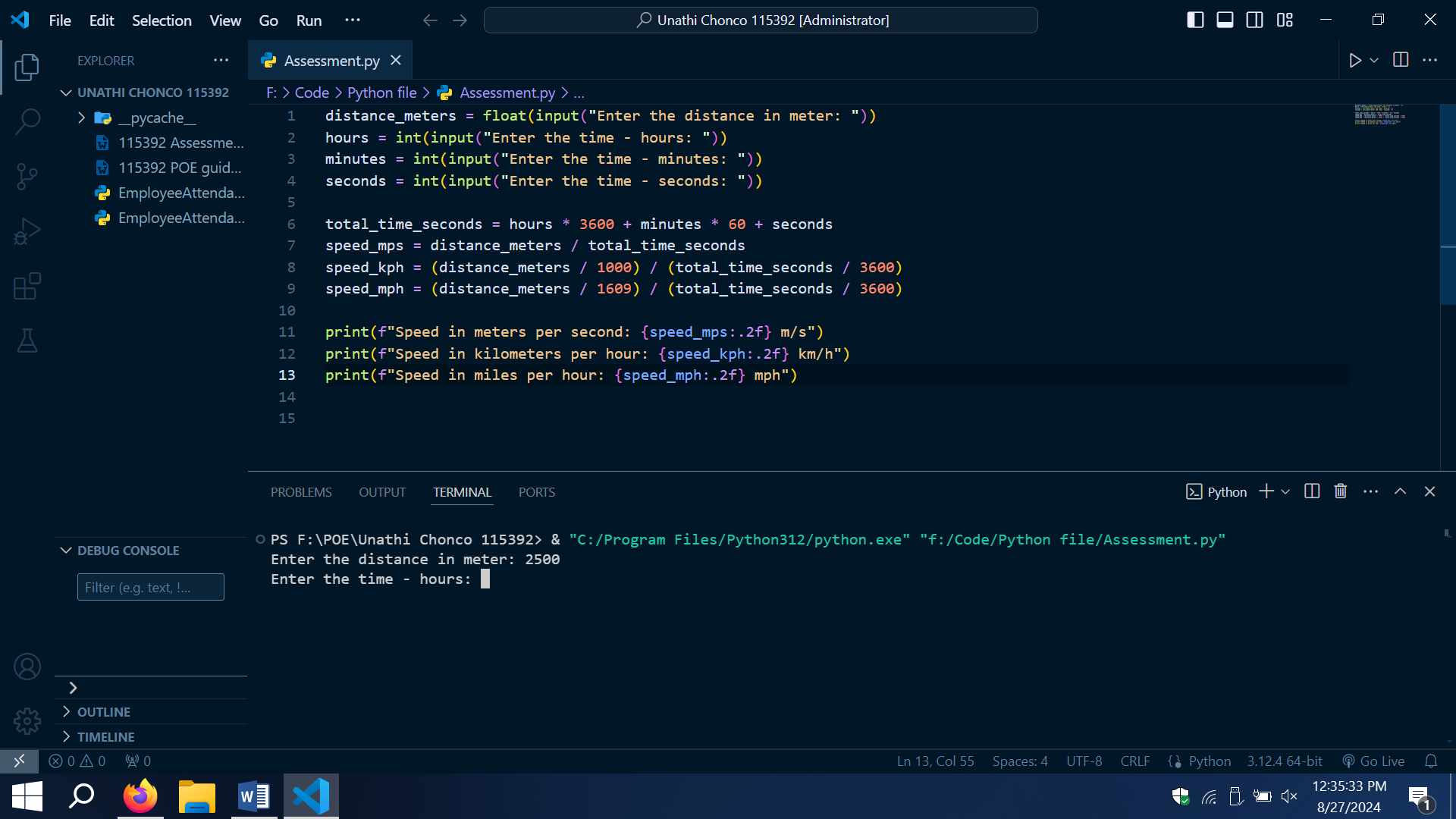
Test Data  
Input the time zone offset to GMT: 256  
Expected Output:   
Current time is 23:40:24

****

**Question 4**

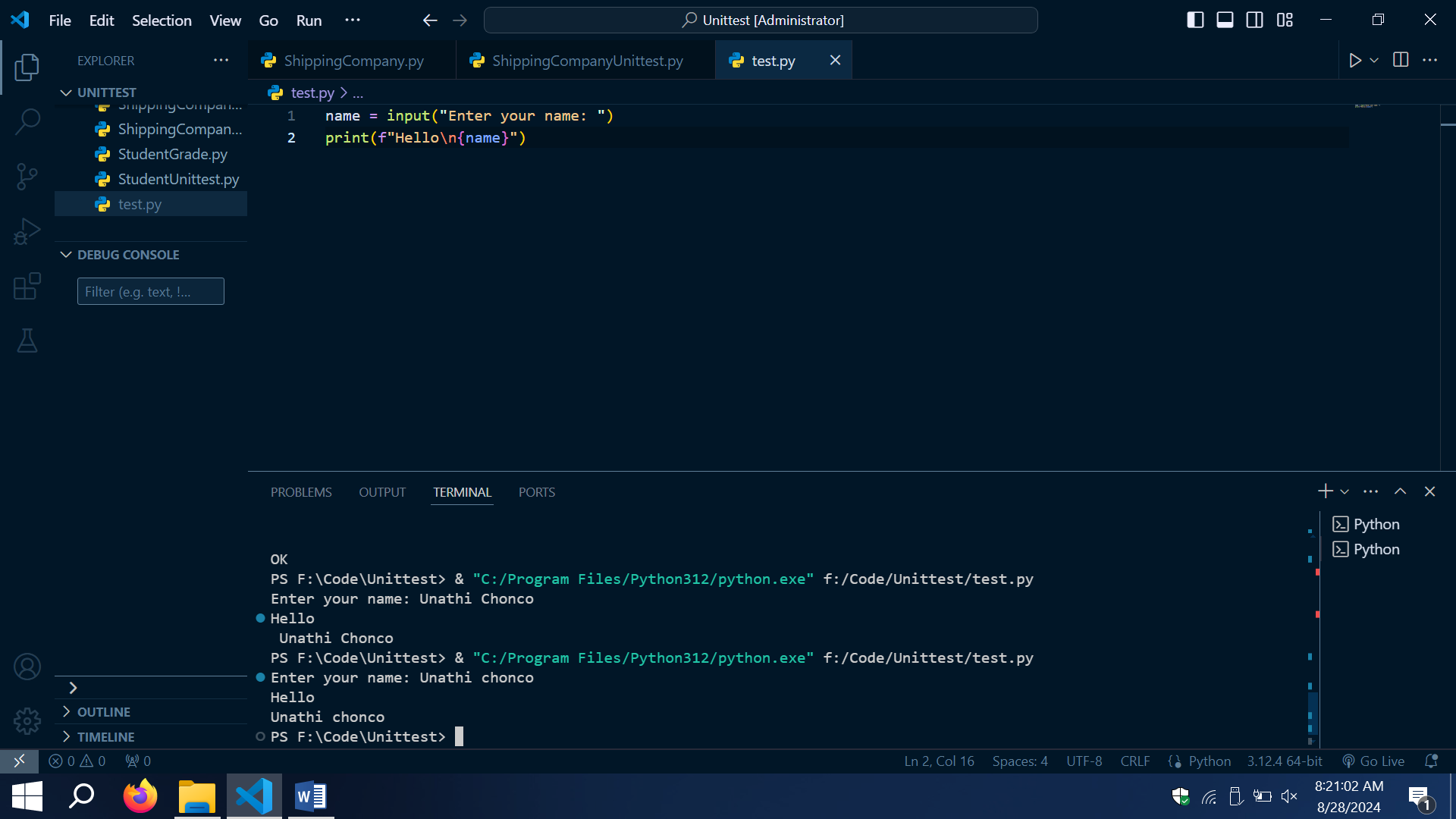
Write a Java program to takes the user for a distance (in meters) and the time was taken (as three numbers: hours, minutes, seconds), and display the speed, in meters per second, kilometres per hour and miles per hour (hint: 1 mile = 1609 meters).

Test Data  
Input distance in meters: 2500   
Input hour: 5   
Input minutes: 56  
Input seconds: 23  
Expected Output:   
Your speed in meters/second is 0.11691531   
Your speed in km/h is 0.42089513   
Your speed in miles/h is 0.26158804



**Question 5**

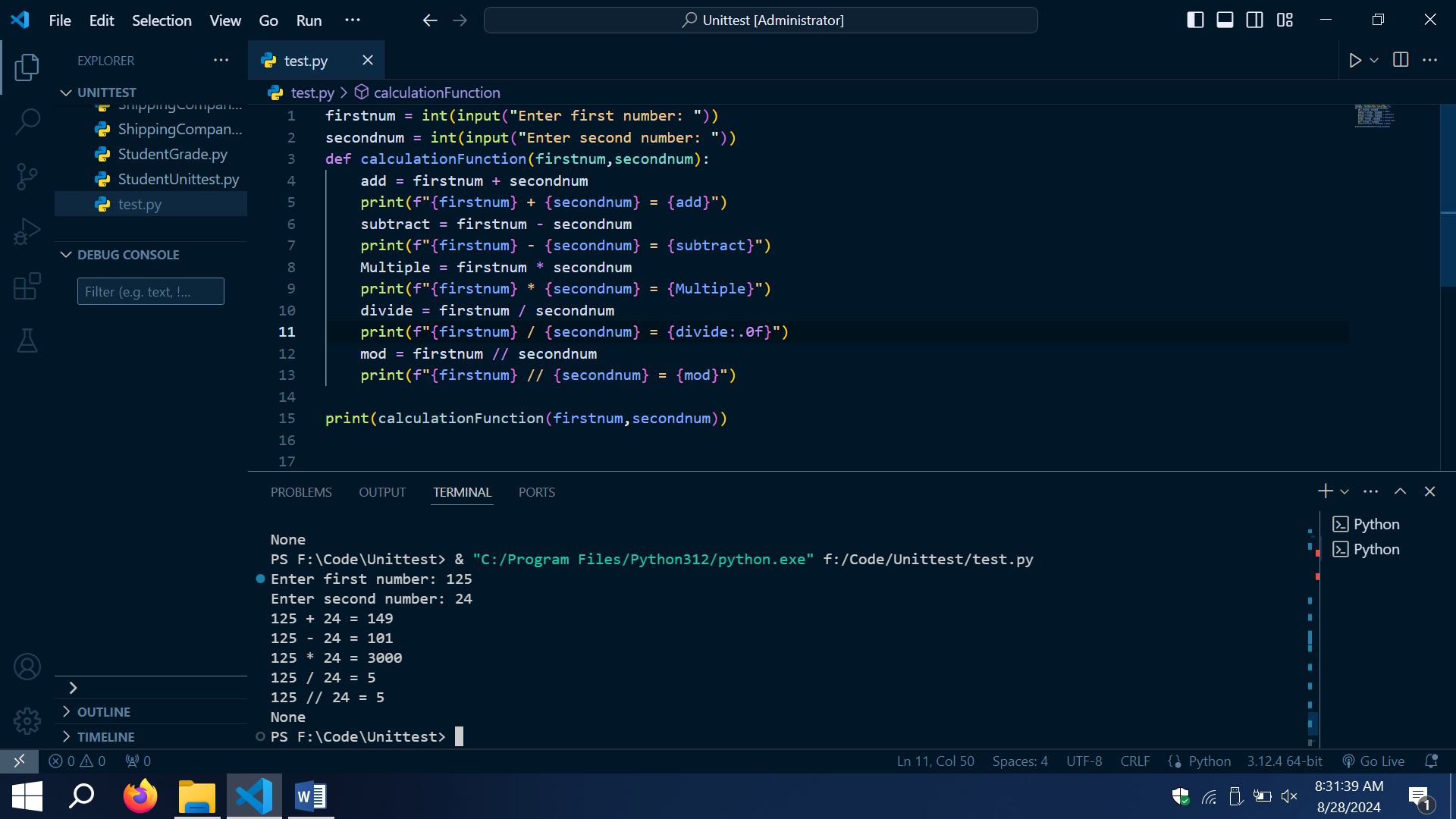
Write a Java program to print 'Hello' on screen and then print your name on a separate line.    
Expected Output:   
Hello   
Alexandra Abramov

****

**Question 6**

Write a Java program to print the sum (addition), multiply, subtract, divide and remainder of two numbers.    
Test Data:   
Input first number: 125  
Input second number: 24

Expected Output:   
125 + 24 = 149  
125 - 24 = 101  
125 x 24 = 3000  
125 / 24 = 5  
125 mod 24 = 5



SECTION B: SUMMATIVE ASSESSMENT

Answer the following questions

**Question 1 (SO 2, AC 1, AC 2, AC 3, AC 4)**

You are required to assist a small business in your community to solve any of its business problems using a computer program.

1. Identify and describe the problem that the small business is facing. (4)

The small business is facing a problem tracking employee attendance and making correct base on the total hours worked

Employee need to manual clock-in and clock-out from the office

A person have to manual count the total hours of an employee and manual calculate the their total pay wage and pay the employee base of those total hours per hourly wage

1. Interpret the problem facing the business and create a plan to develop a computer program solution. The plan must;
2. Propose a description of the problems to be solved by the development of the computer program.

In many small or large businesses, tracking employee attendance and calculate their payroll manual, is a time consuming process. Employee need to manual clock-in and clock-out, which this could lead to error and disputes over hours worked.

The problem to be solved is development of the computer program that:

**Time tracking:**

It will automate the process of tracking employee clock-in and clock-out time reducing the errors tracking the total number of hours worked

Payroll Calculation:

It will automatically calculates the amount to be paid to employees based on their working hours and wage

The overall program will make the process of tracking and paying employee easier and efficient, reducing errors that will cost the business to make incorrect payment

And it will save the company time since everything will be automated

1. Integrate the research of problems in terms of data and functions.

**Data requirement:**

**Employee Data:** Employee ID, Hourly Wage

**Attendance Data:** Clock-in and Clock-out

**Payroll data:** calculate hours worked, total

**Function Requirement:**

**Employee Initialization:**

The employee class will store the employee ID, hourly wage and attendance record

**Clock-in and Clock-out**

The Clock-In function record the time the employee starts work

The Clock-in function record the time the employee finish work

**Calculate Hours worked**

This function will calculate the total number of hours worked by summing time difference between clock-in and clock-out

**Calculate Payment**

This function will calculate payment by multiple the total hours worked by the employee hourly wage and give the total payment of the day

1. Include an evaluation of the viability of developing a computer program to solve the problem identified and compares the costs of developing the program with benefits to be obtained from the program.

The development of such a system is possible to do with programming languages like python and databases. The program can be developed as desktop application or a web based system. Development costs include programmer time, testing and a small budget for software tools or database service

**Costs**

Training employees to use the system

Development of the system and setup

Maintenance and update

**Benefits**

Time Saving since everything will be automated

Will reduce manual entry errors and ensure accurate records

Clear and accurate reporting can improve transparency

Investing in this system will also help when the business grows, the system can scale to handle more employees

1. Choose the best solution to the problem and must document the program features that will contain the capabilities and constraints to meet the defined problem. (15)

The best solution is to develop a python program that tracks employee attendance and calculates payroll

**Program Features:**

User Interface: A simple interface for employees to input their employee ID, clock in and clock out.

Automatic Calculations: The system will automatically calculate hours worked and wages earned, reducing the need for a person to manual calculations.

Display to the employee the time they clock-in and clock-out. And the total hours, they worked and total payment

**Constraints:**

The program must ensure that all data entries (clock-ins and clock-outs) are time stamped accurately

The system should be designed to work across multiple devices if it's web-based, or be easy to install if it's a desktop application

**Question 2 (SO 2, AC 1, AC 2, AC 3, AC 4)**

Using the plan that you developed in question 1 above, you are required to design a computer program. The computer program design must meet the following specifications;

* Incorporate development of appropriate documentation and is desk checked
* Include program structure components
* Include program logical flow components
* Include data structures and access method components

As evidence, you must save the program design on a disk and attach to your POE. (10)

**Step 1: Define a problem**

* Write a computer program that will help a small business to track employee attendance(clock-in and clock-out) and calculate the total hours and make payment on their total hours and hourly wages

**Step 2: Develop Overall Program Structure**

* Input: Employee ID
* Output: The Employee ID, Clock-in and Clock-out time, Total hours worked, Total wage

Main Components

* Clock-in and Clock-out Function
* Calculate Total Hours Function
* Calculate Payment Function

High-Level-Flowchart

Start

Run the program

Display Main Menu

Options: Add\_Employee, Clock-In, Clock-Out, Generate Report, Exit

User Selects an Option:

User choice the program proceeds to corresponding step

Add Employee:

Input Employee Id, Name

Store Employee Information

Return to main menu

Clock In

Input employee ID

Check if employee ID exist

If Yes employee clock in, if No display an error message

Return to main menu

Clock Out

Input employee ID

Check if employee ID exist

If Yes employee clock Out, if No display an error message

Return to main menu

Generate Report

Calculate total hours worked and payment for each employee

Display the report

Return to main menu

Exit Program

End program

**Step 3: Break Down into Modules**

* Module 1: It provide the interface for employees to enter Employee ID
* Module 2: This module handles the recording of Clock-in and Clock-out times, ensuring that the data stored accurately for each employee.
* Module 3: Calculates the total hours worked by each employee and payment based on hourly wage

**Step 4: Design Module Interfaces**

Employee attendance Functions

* Clock-in and Clock-out Function
* Calculate Total Hours Function
* Calculate Payment Function

Main Program

* Inputs: Employee ID
* Go through Functions:
* Clock-in and Clock-out Function
* Calculate Total Hours Function
* Calculate Payment Function

Output: The Employee ID, Clock-in and Clock-out time, Total hours worked, Total wage

**Step 5: Implement Module**

User interface Development

* Basic interface for employee enter their employee ID

Attendance tracking Development

* Implement the clock-in and clock-out functions, ensure accurate timestamp

Payroll Calculation Development

* Development the hour’s calculation and payment calculation functions

**Step 6: Integrate Module**

* First the employee Enter employees
* Second the employee ID will go through all the functions
* Third print out the result
* This will be combine to a single main program

**Step 7: Test and Refine**

* Test all modules to ensure they work together seamlessly
* Validate the accuracy of hours worked and payroll calculation.

The assessor/facilitator must complete the attached evaluation checklist.

**Question 3 (SO 3, AC 1, AC 2, AC 3, AC 4)**

You are now required to create a computer program that implements the design. Take note of the following;

* The creation must include coding from design documents.
* Names created in the program must describe the purpose of the items named.
* The creation includes conformance with design documentation.

As evidence, you must save the computer program on a disk and attach to your POE (15)

The assessor/facilitator must complete the attached evaluation checklist.

**Question 4 (SO 4, AC 1, AC 2, AC 3)**

1. Develop and attach a testing strategy. (6)

**Objective:**

The objective of the testing strategy is to ensure that the employee attendance and payment system functions as intended and meets the specified requirements. The testing process will focus on verifying the accuracy of the functions (clock-in/clock-out, payment calculations, and report generation)

**Scope:**

**Objective**: Verity that individual component work correctly in isolation

**Testing :**

Testing the clock in, clock out, total\_hours\_worked, calculate\_payment and report function within the Employee class

Testing the add\_employee, employee\_clock\_in, employee\_clock\_out and generate\_reports functions within the Employee System class

**Feature:**

Employee clock-in and clock-out functionality.

Calculation of total hours worked by employees

Calculation of employee payment based on hourly wage

Generation of daily reports summarizing employee attendance and payment

**Tool:** unit testing framework

**Testing Methods:**

**Black Box testing:** focuses on testing the functionality of system without knowledge of the internal code structure. Which is suitable for testing the user interface/ input/output of the system

**White Box testing:** focus on testing the structure an d logic of the code.Suitable for testing individual methods within the “Employee” and “Employee\_System” class

**Testing techniques:**

**State transition testing:**

Test the behaviour of the system as it transitions from one state to another.

For example, test transitions from “clocked in” to “clocked out” and verify that the system

1. You are required to demonstrate how to test the computer program that you have developed in question 3 above. This must be done in the presence of the assessor/facilitator. (10)
2. Record the testing results. Attach the testing results in your POE. (6)

**Question 5 (SO 5, AC 1, AC 2, AC 3)**

You are required to implement the program in order to meet the needs of the small business. Follow the guidelines below;

1. Explain how you shall check that the implementation complies with user expectations. (6)  
   **1. Unit Testing:**   
   I will use unit tests to verify each function in the program, clock-in function, clock-out function, total\_hours\_worked function and calculate\_payment function. Ensure that these functions produce correct outputs for various scenarios

**Integration Testing:**

Test how different components of the system work with each other. For example check to ensure that an employee who clock-in and out is correctly reflected in the total hours worked and the payment calculation is correct.

**Compare the program output with the expected output to ensure compliance:**

**Add Employee Function:**

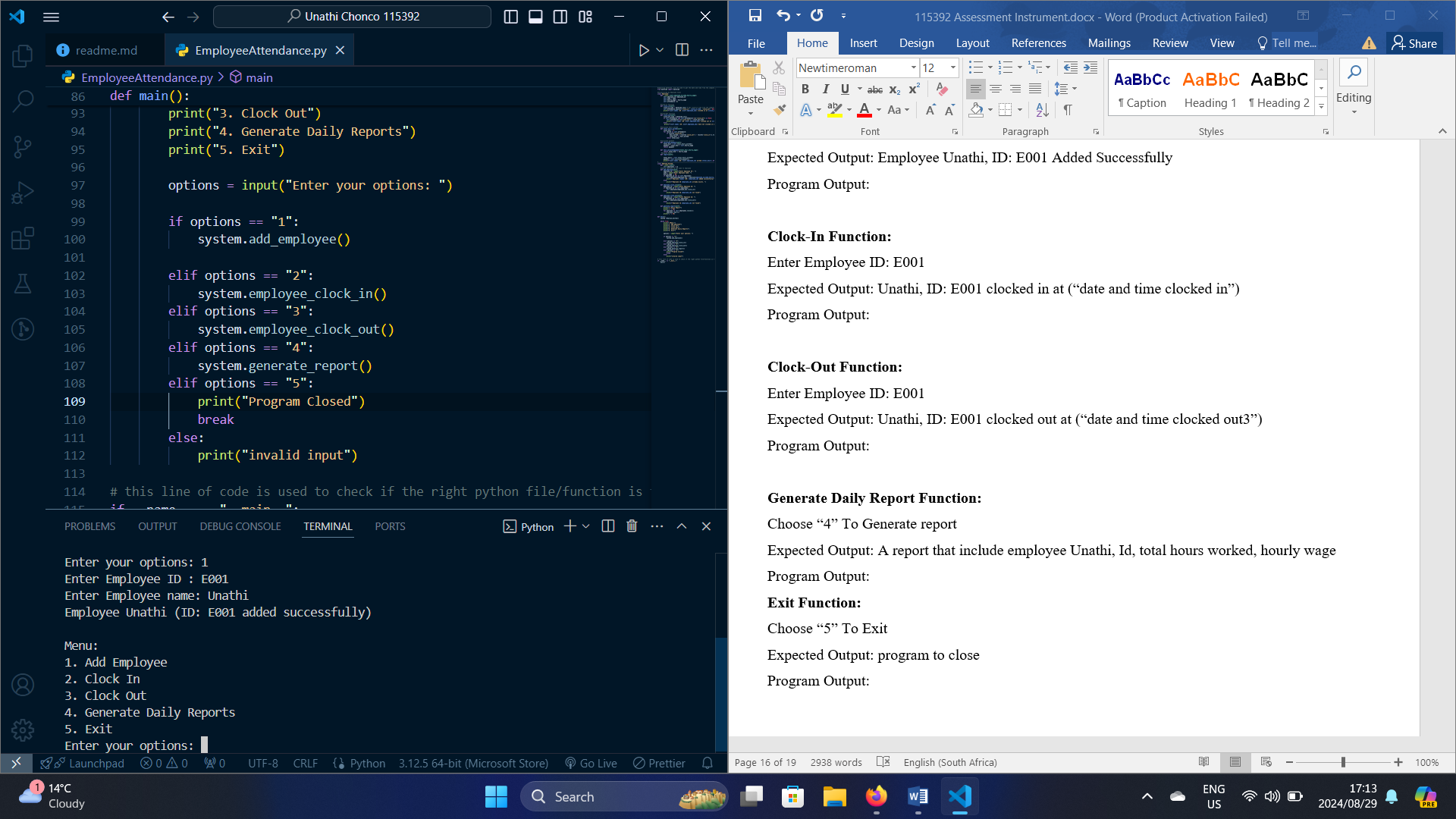
Choose “1” To Add Employee

Enter Employee ID: E001

Enter Employee Name: Unathi

Expected Output: Employee Unathi, ID: E001 Added Successfully

Program Output:



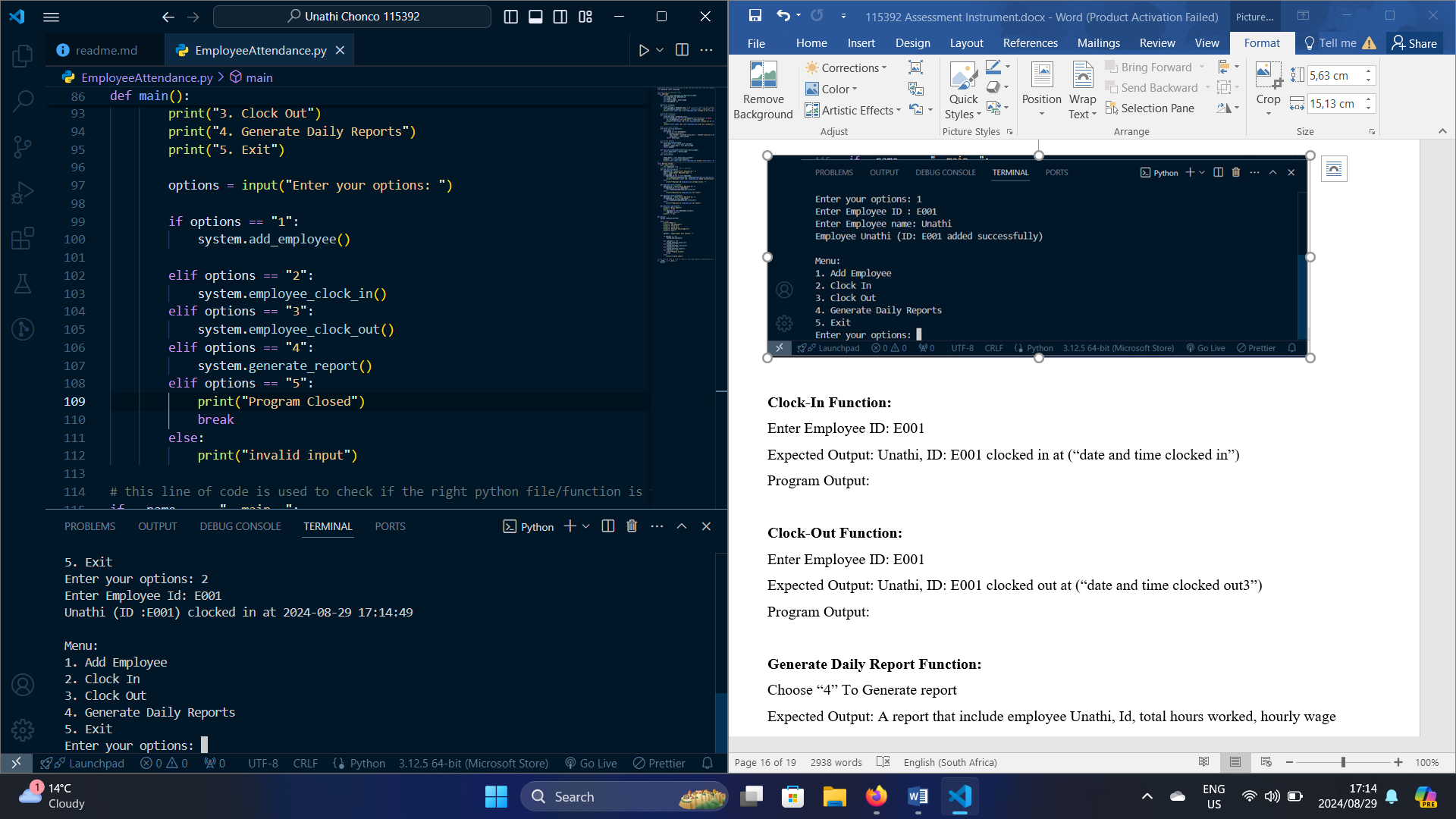
**Clock-In Function:**

Choose “2” To Clock-In

Enter Employee ID: E001

Expected Output: Unathi, ID: E001 clocked in at (“date and time clocked in”)

Program Output:



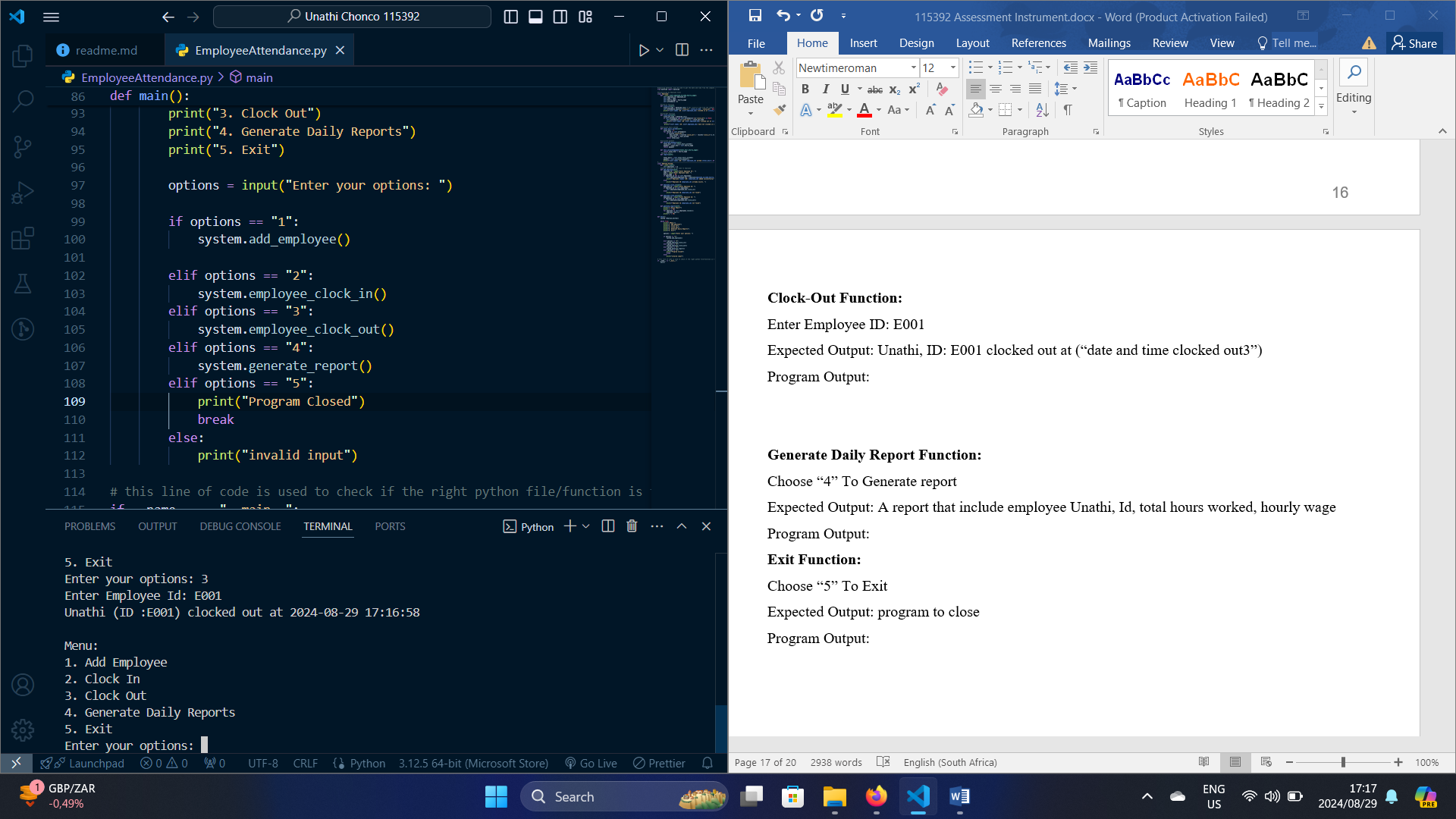
**Clock-Out Function:**

Choose “3” To Clock Out

Enter Employee ID: E001

Expected Output: Unathi, ID: E001 clocked out at (“date and time clocked out3”)

Program Output:

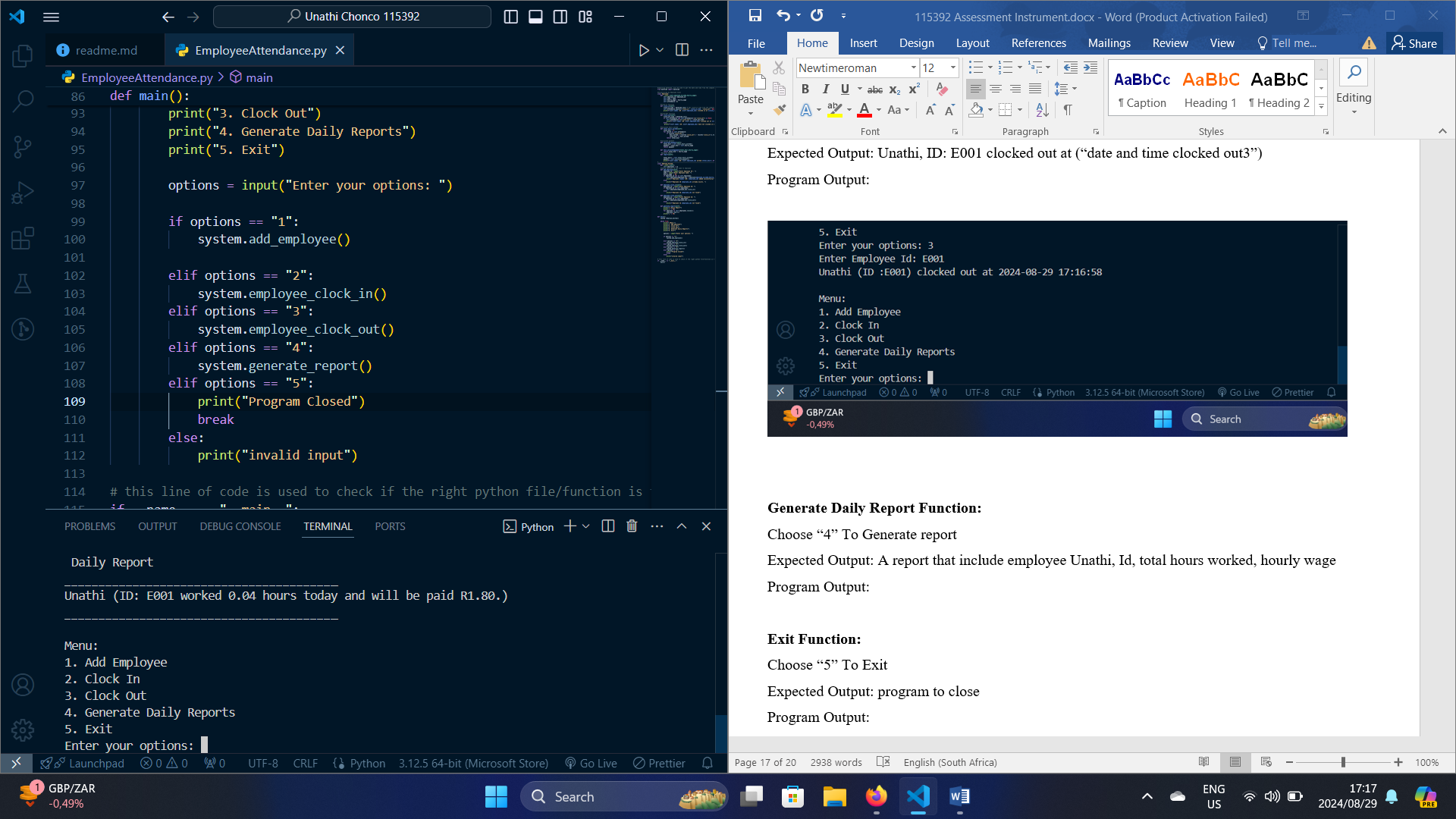


**Generate Daily Report Function:**

Choose “4” To Generate report

Expected Output: A report that include employee Unathi, Id, total hours worked, hourly wage

Program Output:

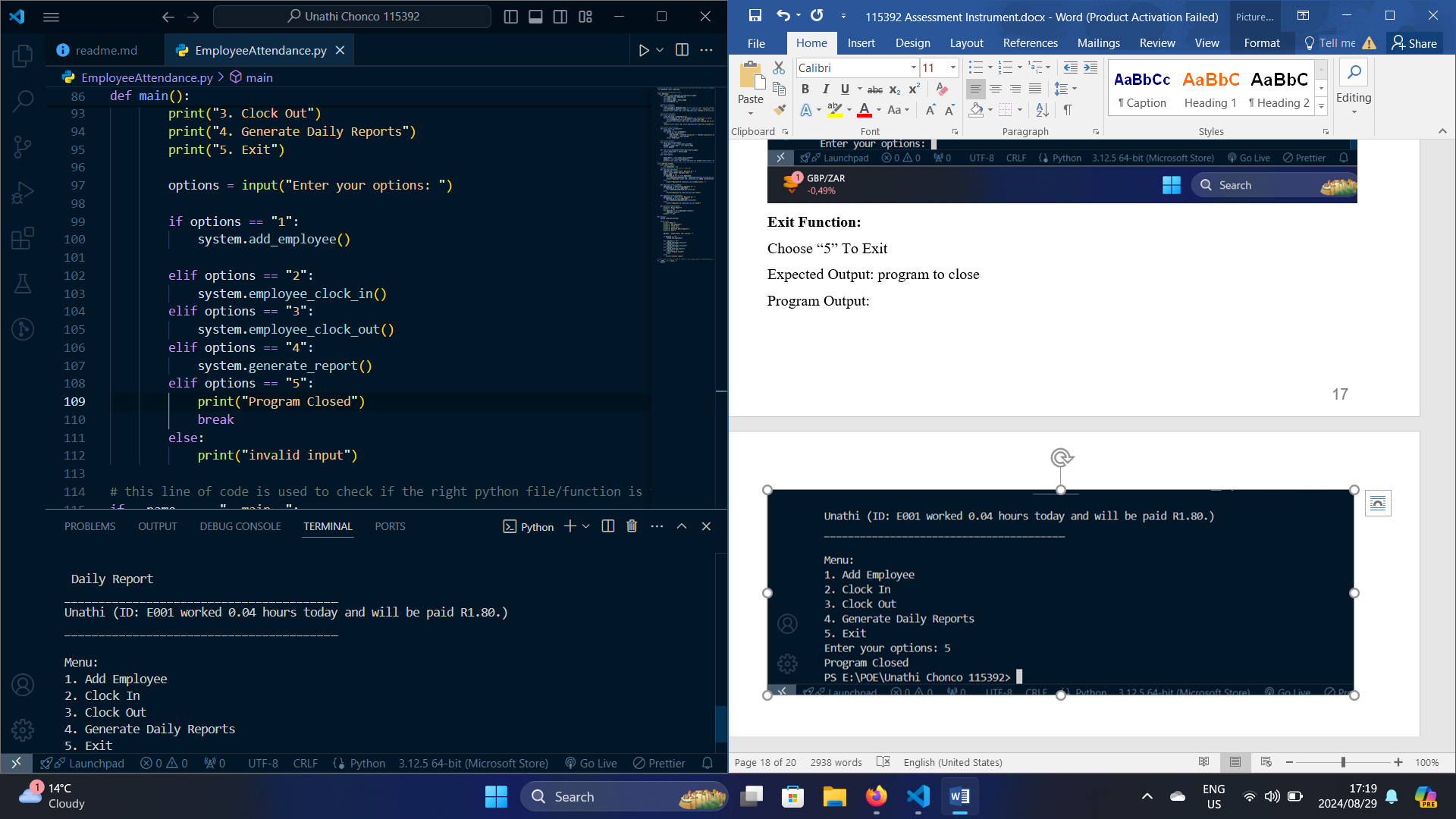


**Exit Function:**

Choose “5” To Exit

Expected Output: program to close

Program Output:



1. Develop a plan for the installation process of the program. (5)

**Prepare the installation Package:**

Python Script   
Readme file

Must be in one package to ensure easy installation

**Compatibility Check:**

Test the installation package on different systems that the small business. Ensure

Compatibility with the target operating system.

**Create an Installation Guide:**

Develop a step-by-step guide for installing the program.

System requirements

Step to install Python if not already

How to run the program

**Installation on Target System:**

Assist the small business owner in installing the program on the target. Ensure the process is smooth and address any issues that arise during installation

**Test the Program:**

After installing, run the program on the target system to verify that it functions as expected

1. Develop a training plan for the small business owner and the users of the system. (8)

**Introduction to the program:**

The Employee Attendance Program purpose is to simplify the management of employee attendance and automate payroll calculations

Features:

Clock-In/Clock-Out functionality: Employee can easily record their working hour by using the clock-in and clock- out feature.

Automated Payment: The calculate payments based on the total hours worked and employee’s hourly wage.

Daily Reports: the system generates daily reports that summarize the total hours worked by each employee and corresponding payment due.

**How to use the program**

Clock-In: Employees start their worked by clocking in the system.

Clock Out: At the end of the workday, employees clock out, recording their end time.

Payment Calculation: The system automatically calculates the payment based on total hours worked and the hourly wage.

Add employee: Using the “Add Employee” option to input new employee data, including their ID, name, and hourly wage.

Generate report: At the end of each day employee can generate a report of the hours their worked

And the their hourly payment

**How to Perform Routine Maintenance**

Back Up the program data to prevent loss of information.

Regularly check that the system clock is accurate, as the program relies on it

**How to customize the program feature**

Wage can adjust are necessary, the program allows you to modify the hourly wage for individual employees directly in their records

Report generation function can be adjusted to include additional information

**Question 6 (SO 6, AC 1, AC 2, AC 3, AC 4)**

Develop a documentation for the computer program. The documentation must;

1. Include annotation of the program with a description of the program purpose and design specifics

**Program Purpose**

The purpose of this program is to manage employee attendance, by tracking Employee Clock-In and Clock-out times, calculate the total hours worked and determine the payment based on their hourly wage. The program is designed to help process of employee time management and payroll calculation for small business to be easier and quicker.

**Design Specific**

* Employee Class: Has six functions use to capturer Employee Details including name, hourly wage and attendance records and Calculate employee payment and hours worked

Function:

Clock In function: store the time the employee started worked and display is to the employee

Clock Out function: store the time the employee Finish worked and display is to the employee

Total Hours worked function: calculate the total hours worked by checking if the employee has clocked out if the employee clock out it will take the clock in and clock out calculate the total hour between the clock in and clock out to get the total hour worked

Calculate payment function: calculate total payment based on total hours worked and hourly wage

Report function: Layout of how to print the daily report

* Employee System Class: act as management system, allowing for adding employees, handle clock-ins and clock-outs and generate reports

Functions:

Add\_employee function: add employee name, id and assign hourly wage.

Clock-In and Clock-Out functions: check whether the employee has clock-in or clock-out

Generate report function: Display the employee report which include employee id name, clock-in and clock out

* Main function: Provide a simple command line interface for employees to interact with the program to add, manage clock-in and clock-out and generate daily reports

1. Include the layout of the program code including indentation and other acceptable industry standards

**Program Code layout**

Modular Design this program is designed in a modular fashion with each class and function performing a specific task.

1. Include internal and external documentation with a level of detail that enables other programmers to analyse the program

Internal documentation

In my python file I have comments that provide guideline for my code

External documentation

I have created a readme that provide the user with guideline to on how the program works and what are the program requirement for it to run, how to use the program step but step.

1. Reflect the tested and implemented program, including changes made during testing of the program.

**Change Made During testing:**

During testing they were no changes made. All of the main function ran ok after testing multiple time.

(15)