Name: MD Ariful Islam

Email: arif\_9011@yahoo.com

Hangman project in python

Introduction: This is a simple Hangman game using Python programming language in Python.

## Milestone 1 Task1

Q: Set up GitHuband environment for the project?

Ans: install VS code, learn command line, file manipulation, set up Github account and create Repo hangman.We will use GitHub to track changes our code and save them online in a GitHup repo.

## Milestone 2 Task1

Q: Define the list of possible words?

Ans:

Step 1: I have created list containing the name of 5 favourite fruits in python.

Step 2: I have assigned this list to a variable called word\_list.

Step3: I did print out the created list to the standard out.

Code:

```python

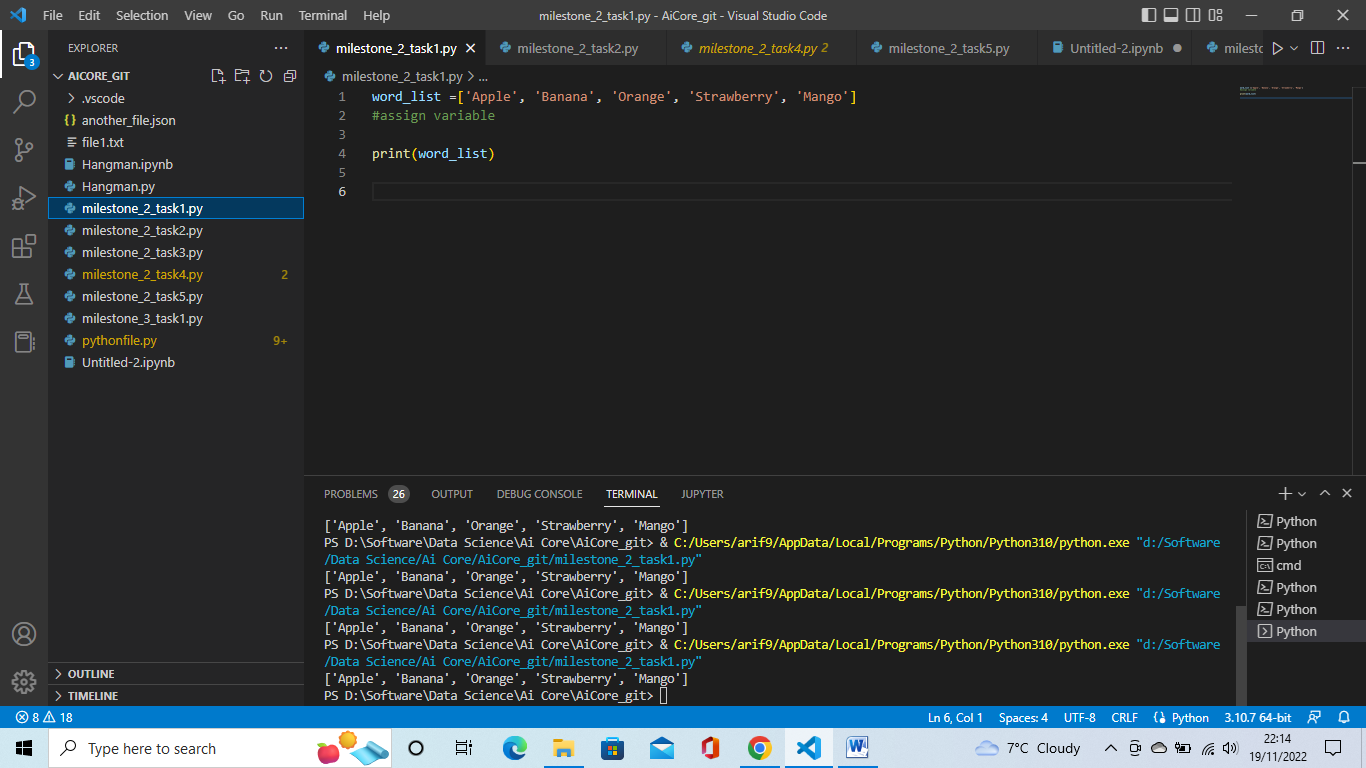
word\_list=['Apple', 'Banana', 'Orange', 'Strawberry', 'Mango']

#assign variable

print(word\_list)

‘’’

Screenshot:



## Milestone2 Task2

Q: Choose a random word from the list?

Ans: I have used random module. The random module is one of python’s built-in modules. It has a choice method which returns a random item from a given sequence.

Step1: I have created new file in python.

Step2: I have written import random on the line

Step3: I have created random.choice method and passed the word\_list variable into the choice method.

Step4: I did assign the randomly generated word to a variable called word.

Step5: Then I did print out the word to the standard output. And I did run the code several times and I can see word is coming randomly.

Code:

```python

importrandom

word\_list=['Apple', 'Banana', 'Orange', 'Strawberry', 'Mango']

random.choice

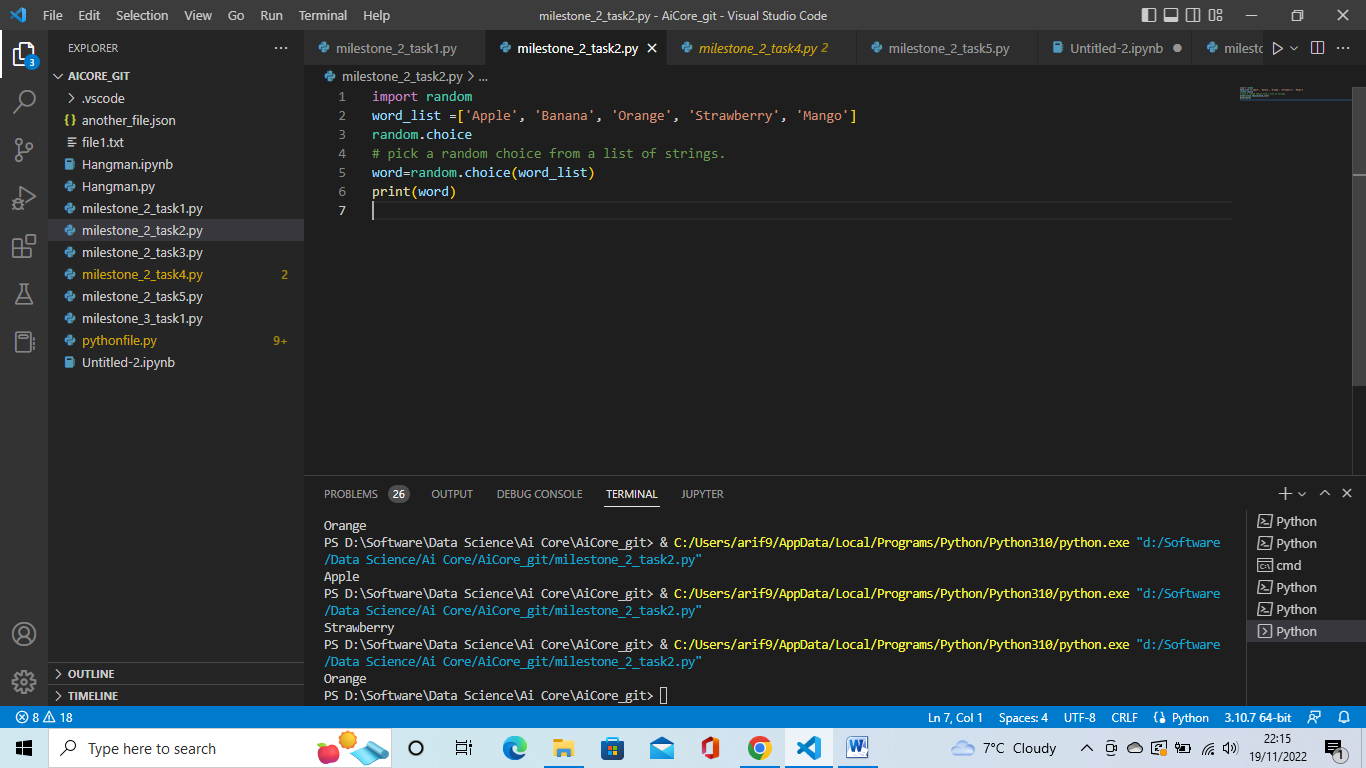
# pick a random choice from a list of strings.

word=random.choice(word\_list)

print(word)

‘’’

Screenshot:



## Milestone 2 Task3

Q:Ask the user for an input?

Ans:

Step1: I have used input function for taking input from the user. Ask the user to enter a single letter.

Step2: I did assign the input to variable called guess.

Code:

‘’’ python

guess=input("Enter a single letter :  ")

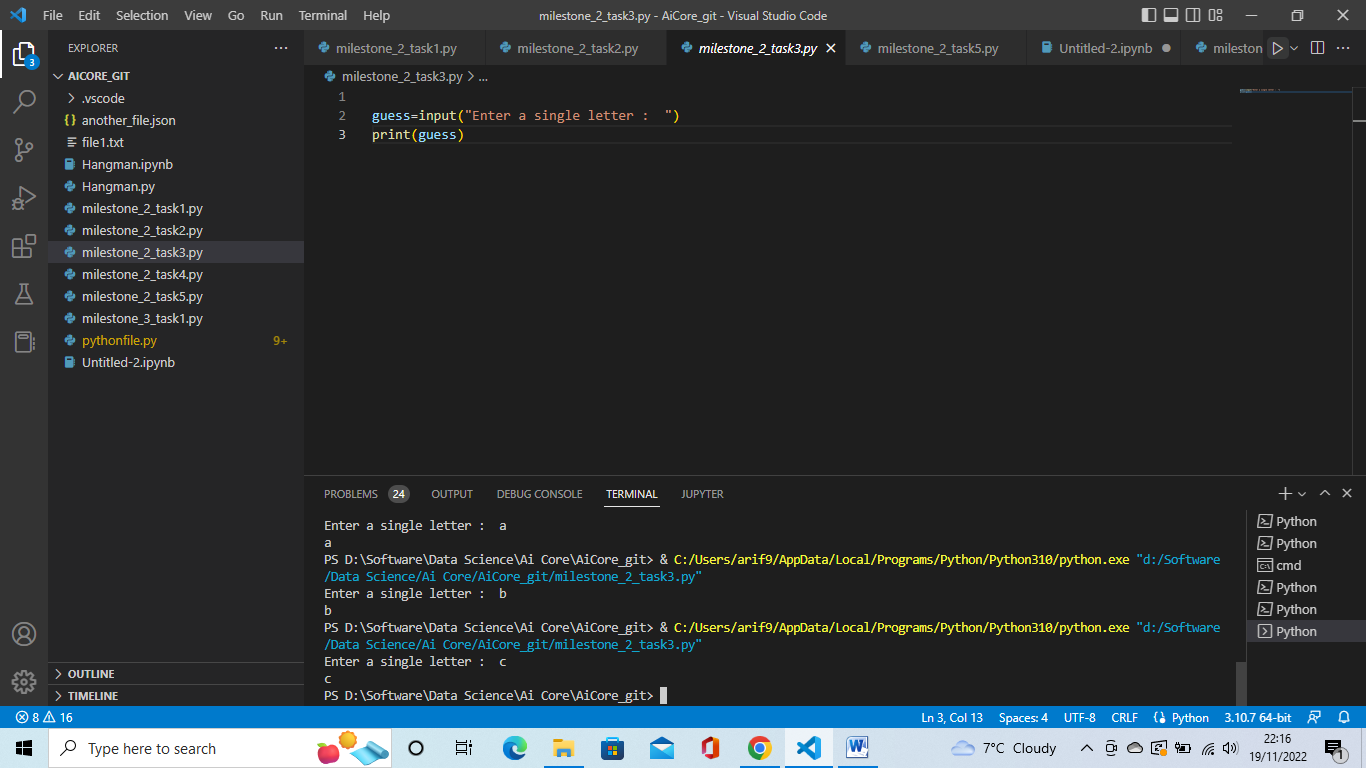
print(guess)

Screenshot:

##Milestone 2 Task 4

Q: Check that input is a single character ?

Ans: Step1: I have if statement and in the input item is equal to 1 and the input is an alphabet.



##Milestone 2 Task4

Q: Check that the input is a single character?

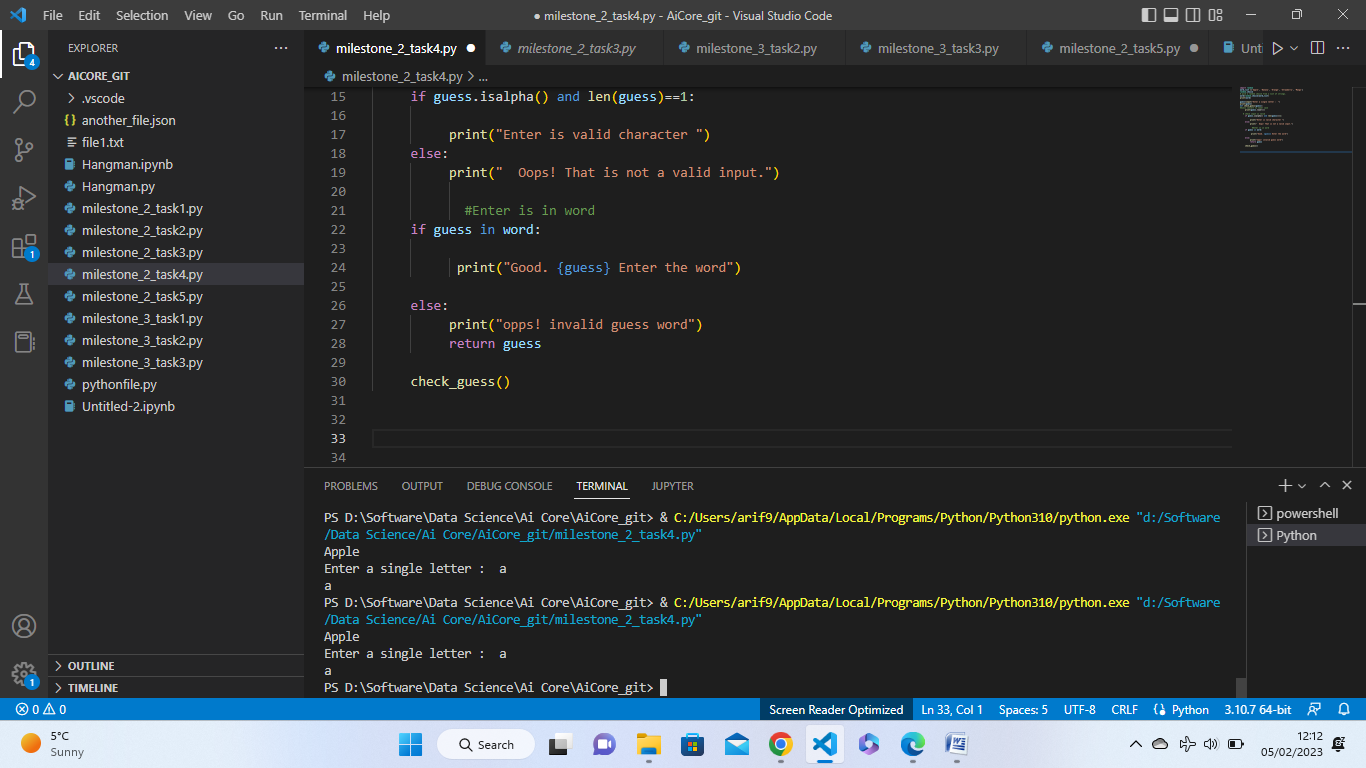
Ans:

Step1: I have created if statement input guess is equal to 1 and the input is an alphabet.

Step2: If the if statement is true then print good guess.

Step3: if the if statement is not true then else block print opps that is not a valid input.

Screen shot:



##Milestone 3 task1

Q: Iteratively check if the input is valid guess

Ans:

Step1: Create a while loop and set condition is true.

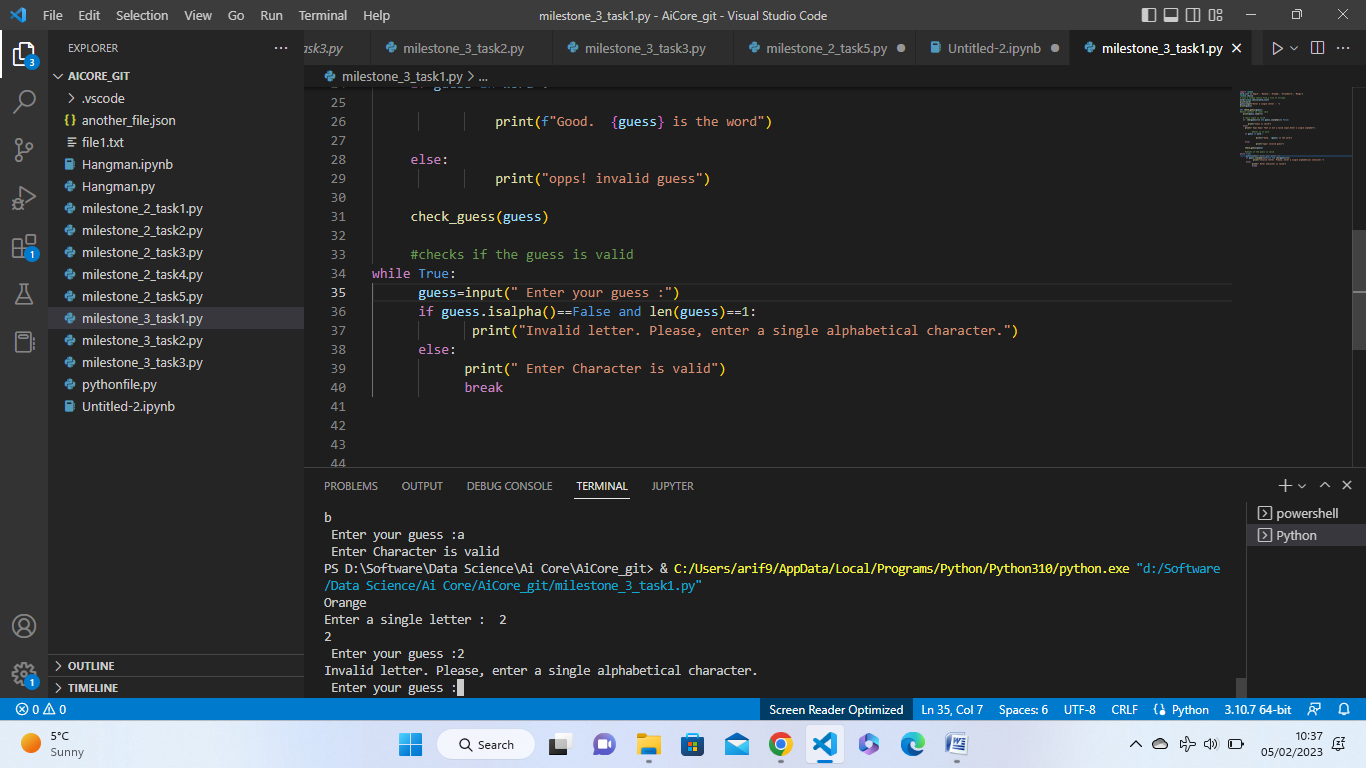
Step2: Ask for user assign variable is called “guess”.

Step3: Check the condition guess is a single , alphabet character.

Step4: If the guess passes the checks then break out of the loop.

Step5: if the guess does not pass the checks then print “ Invalid letter, enter a single alphabetic Character.

Screen shot:



##Milestone 3 task2:

Q: Check whether guess is in the word?

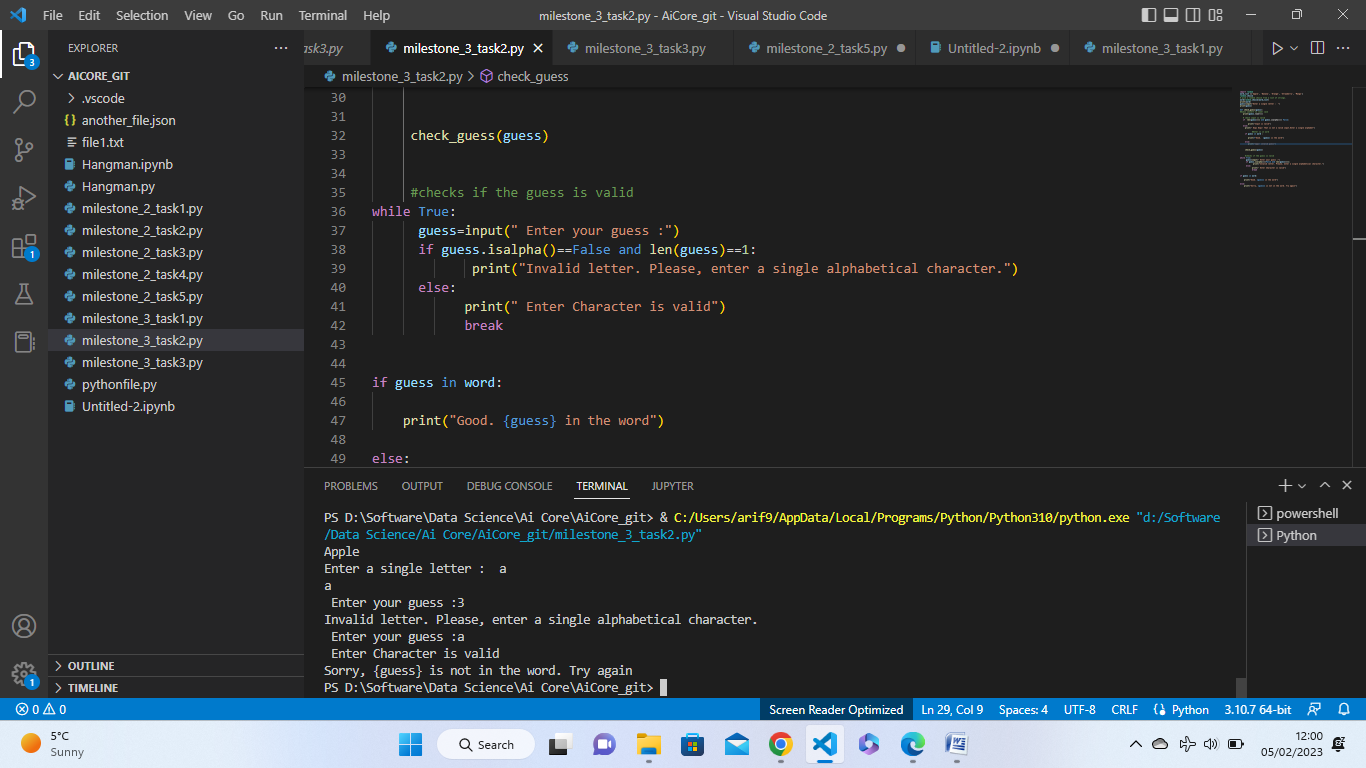
Ans:

Step1: I have created an if statement for checking if the guess is in the word

Step2: if the if statement is guess then print “Good guess! {guess} is in the word

Step3: and I have created else book and print the message “sorry guess is not the word, try again”

Screen Shot:



##Milestone 3 task3

Q:Create the function to run the checks?

Ans: Step1: Define a function called check\_guess() and pass a guess as parameter.

Step2: Convert the guess into lower case.

Step3: Move to code check if the guess is in the word into this function into ask\_for\_input()

The ask\_for\_input function

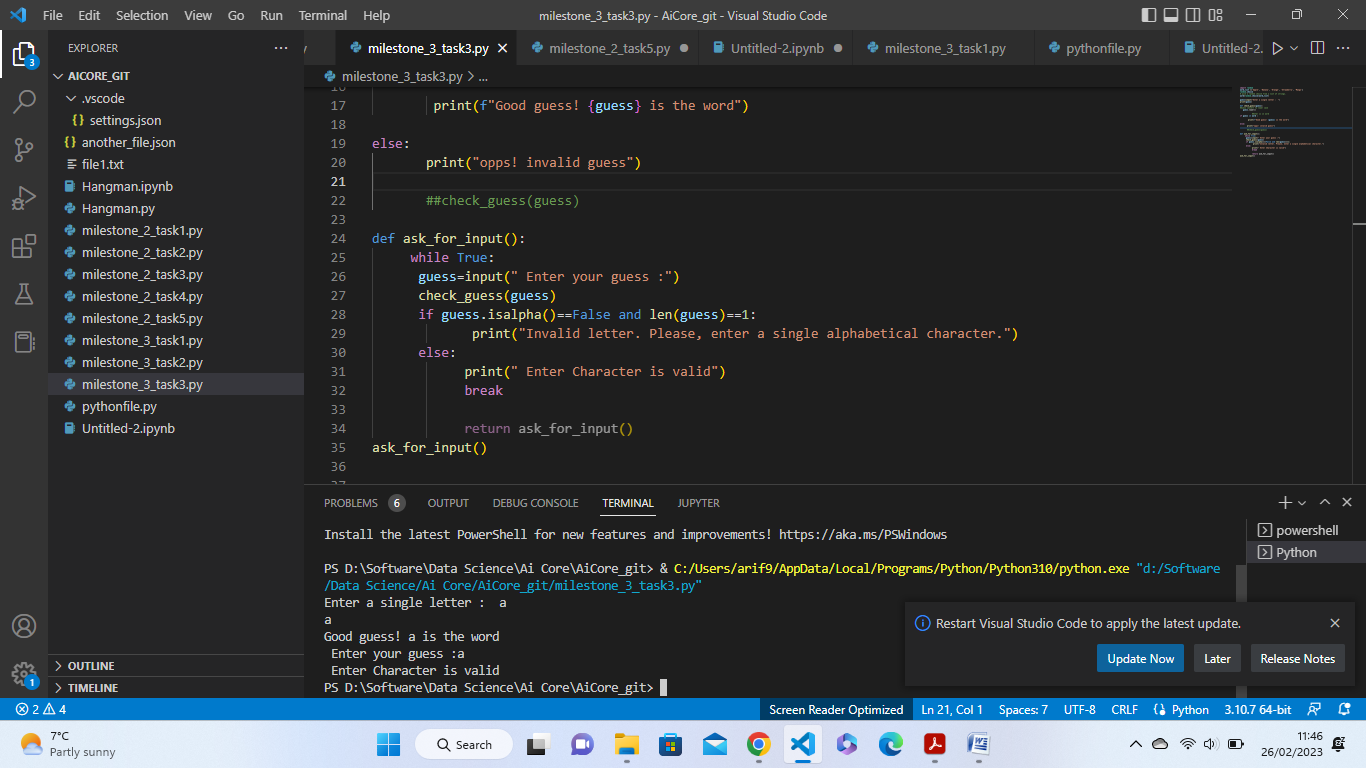
Step1: Define a function called ask\_for\_input()

Step2: Move that code that wrote into Iteratively check if the input is valid guess task into this function block.

Step3: Outside the while loop, within this function call the check\_guess function to check if the guess is in the word and pass guess as a argument to the method.

Step4: Outside the function, call the ask\_for\_input() to test your code.

Screenshot:



##Milestone 4 task1:

Q: Create the Class

Ans:

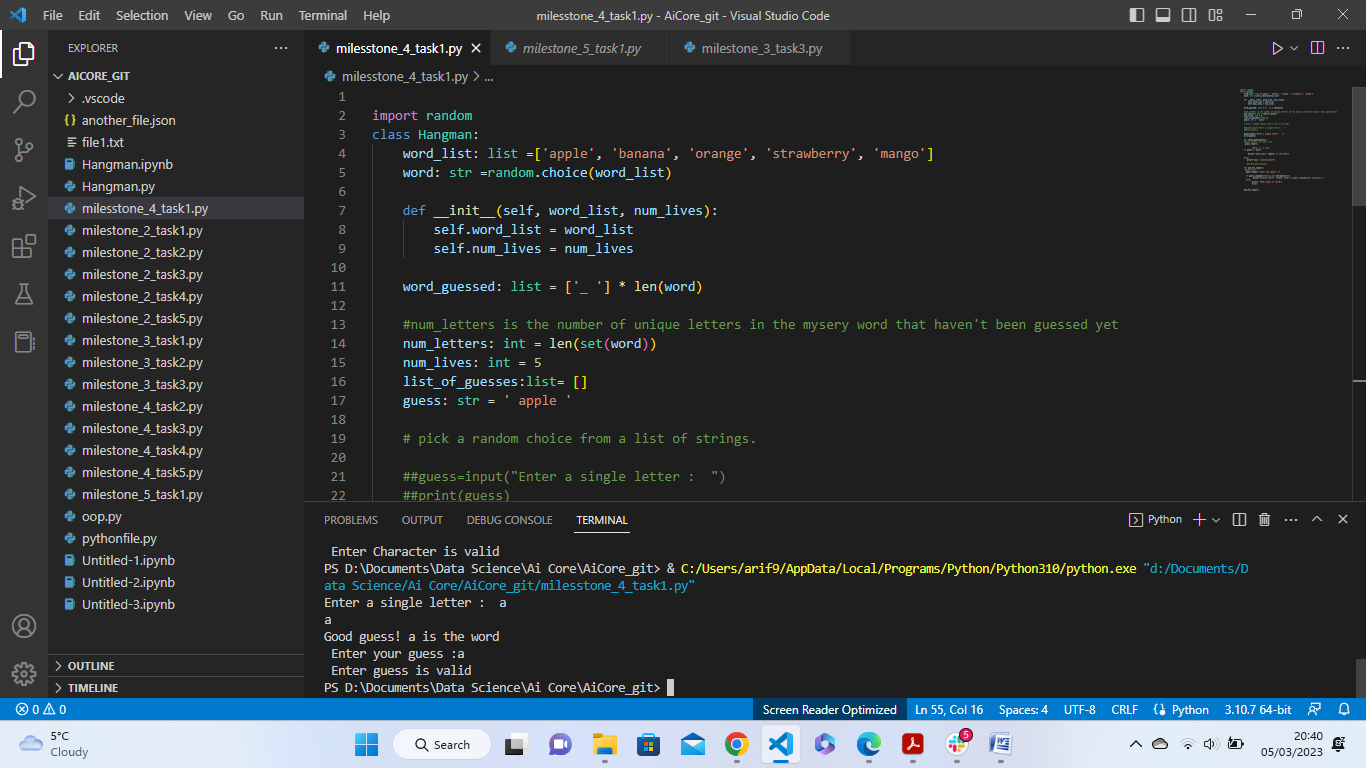
Step1: I have created game class is name Hangman.

Step2: Inside the class, created \_\_init\_\_() method and inside the method pass parameter word\_list and num\_lives and set values num\_lives = 5 .

Step3: Initialisation the attributes

1. The word to be guesses , picked randomly from the word\_list.
2. A list of the letters of the word which letter not yet guessed.
3. Num\_letters: int -The number of Unique letters in the word that have not been guessed yet.
4. Num\_lives: int – The number of lives the player has at the start of the game.
5. Word\_list: list – A list of words.
6. List\_of\_guesses: list – A list of the guesses that have already been tried and I have set to an empty list initially.

Screen Shoot:



##Milestone 4 task2:

Q: Create methods for running check?

Ans: For check\_guess method

Step1: Defined the method check\_guess()

1. I have defined check\_guess method and pass guess as parameter inside the Hangman class. Then converted guessed letter to lower care.
2. If statement that checks if the guess is in the word then print “ Good guess{guess} is in the word”

Step2: Defined method ask\_for\_input.

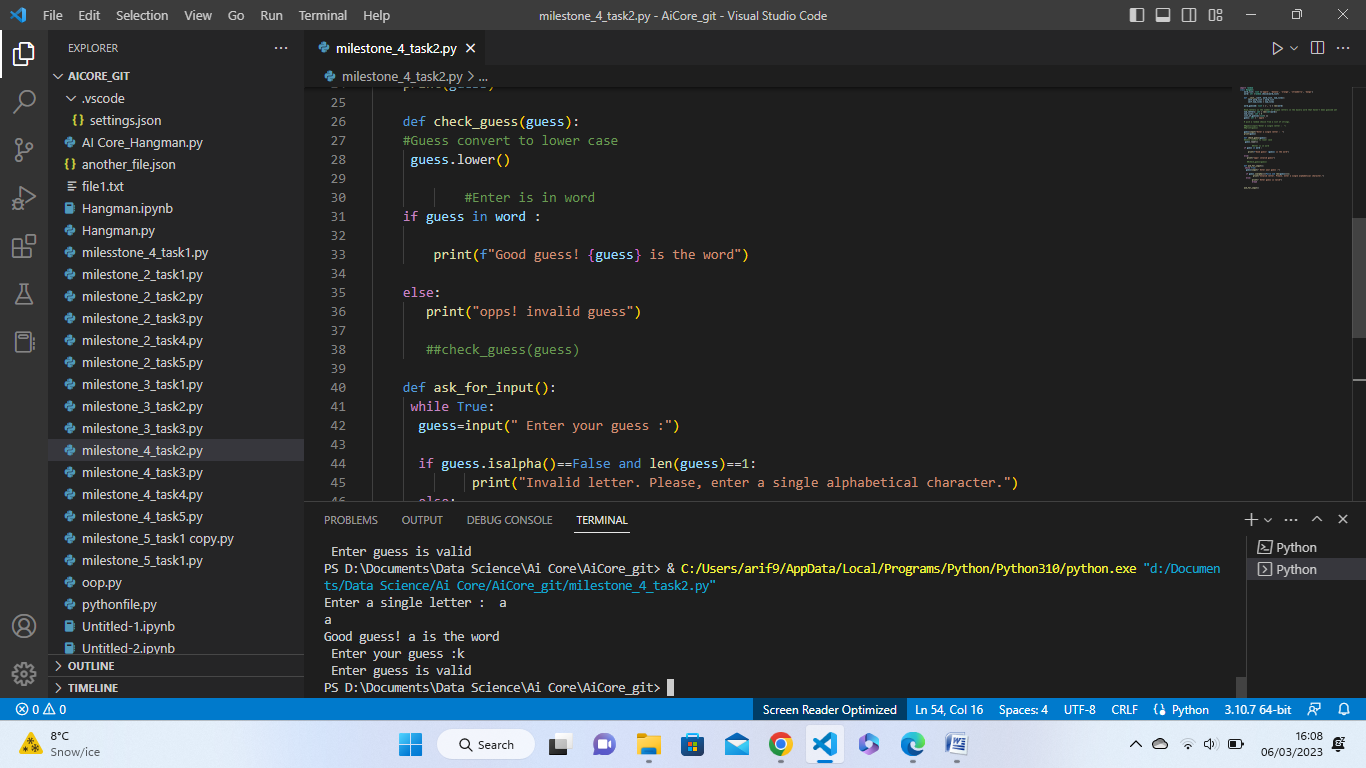
1. I have created While loop and condition put True.
2. I have assigned variable called guess.
3. If the statement guess is NOT a single alphabetic character
4. Then in the body of the if statement print ”Invalid letter. Please, enter a single alphabetical character”.
5. Then created elif statement guess is in the list\_of\_guesses.
6. Then in the body of the elif statement print” You already tried that letter”

In the else block I have called check\_guess() method and pass guess as a argument

Finally add append the guess to the list\_of\_guess.

Step 3: I have called ask\_for\_input mrthod for test the code.

Screen Shoot:



## Milestone 4 task3:

Q: Define what happened if letter is in the word?

Ans: I have created for loop inside the check\_guess method. Then I have created if statement if block, replace the corresponding “-“ in the word\_guessed with the guess. I did index the word\_guessed at the position of the letter and assign it to the letter. And the outside the for loop , reduce the variable num\_letters by 1.

Milestone 4 task4:

Q: Define what happens if the letter is NOT in the word

Ans: In the check\_guess method , I have created else statement. Within the else block reduce ‘num\_lives’ by 1. Then print a message saying “sorry, {letter} is not in the word and print another message saying “You have {num\_lives}lives left.

Milestone 5 task1:

Q: Code the logic of the game?

Ans:

Step1: I have created the method play\_game(). Then I have created an instance of the Hangman class.

Step2:

I have done this by calling the Hangman class and assign it to a variable called game.

Step3: I have word\_list as arguments to the game object.

Step4:

I have created while loop and set the condition is true. Then in the body of the loop do following step:

1. Check if the ‘num\_lives’ is 0. If it is, that means the game has ended and the user lost. Then print”You lost”.
2. Next, if the check ‘num\_letters’ is greater than 0. In this case, I want to continue the game then I need to call ‘ask\_for\_input’ method.
3. If the ‘num\_lives’ is not 0 and the ‘num\_letters’ is not greater than 0,that means the user has won the game. Then print”Congratulations, You won the game!”

Outside the function, I have called play\_game() for playing game.

Screen Shoot:

