

Assignment – 8

AI Assisted coding

2303A51897

Task 1: Email Validation

PROMPT :

Implement a function to validate email addresses based on the following criteria:

- The email must contain exactly one "@" symbol.

- The email must not start or end with "@" or ":".

- The email must contain at least one ":" after the "@" symbol.

The screenshot shows a code editor interface with several windows open. The main window displays Python code for validating emails. The code includes a function `is_valid_email` that checks if an email has exactly one '@' symbol, does not start or end with '@' or ':', and contains at least one '.' after '@'. It also includes a test class `TestEmailValidation` with methods for valid and invalid emails. The bottom terminal window shows the command `python 2303A51897_5.py` being run, and the output indicates 13 tests ran in 0.002s with a result of 'OK'.

```
File Edit Selection View Go Run Terminal Help
Welcome 2303A51897_5.py x
2303A51897_5.py > is_valid_email
14 def is_valid_email(email):
15
16     if email.count('@') != 1:
17         return False
18
19     if email[0] in "@.:" or email[-1] in "@.:":
20         return False
21
22     if ":" not in email:
23         return False
24
25     pattern = "[A-Za-z0-9._%-]+@[A-Za-z0-9.-]+\.[A-Za-z]{2,}$"
26
27     return re.match(pattern, email) is not None
28
29
30
31 class TestEmailValidation(unittest.TestCase):
32
33     def test_valid_emails(self):
34         self.assertTrue(is_valid_email("test@example.com"))
35         self.assertTrue(is_valid_email("user.name@domain.com"))
36         self.assertTrue(is_valid_email("abc123@gmail.in"))
37
38     def test_invalid_emails(self):
39         self.assertFalse(is_valid_email("testexample.com"))
40         self.assertFalse(is_valid_email("test@example..com"))
41         self.assertFalse(is_valid_email("@example.com"))
42         self.assertFalse(is_valid_email("test@.com"))
43         self.assertFalse(is_valid_email("test@com"))
44         self.assertFalse(is_valid_email("test@domain."))
45
46
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS
PS C:\Users\raghav\Desktop\Temp\AI Assistance in coding & c:/Python314/python.exe "c:/Users/raghav/Desktop/Temp/AI Assistance in coding/2303A51897_5.py"
=====
Ran 13 tests in 0.002s
OK
PS C:\Users\raghav\Desktop\Temp\AI Assistance in coding>

```

SESSIONS CHAT + SESSIONS 1 w 1 wk ago Python function with syntax err... Completed in 4s.

Ln 15, Col 35 Spaces: 4 UTF-8 CRLF Python 3.14.2 Auto Go Live

Task 2: Grade Assignment

PROMPT:

```
# Create a function that takes a student's score (0-100) and returns the corresponding grade based on the following scale:
```

- 90-100: A

- 80-89: B

- 70-79: C

- 60-69: D

- Below 60: F

File Edit Selection View Go Run Terminal Help

Welcome 2303A51897_5.py X

```
def assign_grade(score):
    if score >= 90:
        return "A"
    elif 80 <= score < 90:
        return "B"
    elif 70 <= score < 80:
        return "C"
    elif 60 <= score < 70:
        return "D"
    else:
        return "F"

class TestGradeAssignment(unittest.TestCase):
    def test_valid_scores(self):
        self.assertEqual(assign_grade(95), "A")
        self.assertEqual(assign_grade(85), "B")
        self.assertEqual(assign_grade(75), "C")
        self.assertEqual(assign_grade(65), "D")
        self.assertEqual(assign_grade(50), "F")

    def test_boundary_values(self):
        self.assertEqual(assign_grade(90), "A")
        self.assertEqual(assign_grade(80), "B")
        self.assertEqual(assign_grade(70), "C")
        self.assertEqual(assign_grade(60), "D")

    def test_invalid_inputs(self):
        self.assertEqual(assign_grade(-5), "Invalid")
        self.assertEqual(assign_grade(105), "Invalid")
        self.assertEqual(assign_grade("eighty"), "Invalid")
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS

PS C:\Users\raghav\Desktop\Temp\AI Assistance in coding> & C:/Python314/python.exe "c:/Users/raghav/Desktop/Temp/AI Assistance in coding/2303A51897_5.py"

Ran 13 tests in 0.0002s

OK

PS C:\Users\raghav\Desktop\Temp\AI Assistance in coding> & C:/Python314/python.exe "c:/Users/raghav/Desktop/Temp/AI Assistance in coding/2303A51897_5.py"

Ran 13 tests in 0.0003s

OK

PS C:\Users\raghav\Desktop\Temp\AI Assistance in coding>

CHAT SESSIONS C Q □ x

Python function with syntax err... Completed in 4s. 1 wk ago

2303A51897_5.py

Describe what to build next

Launchpad

Task 3: Sentence Palindrome Checker

PROMPT:

```
# Implement a function to check if a given sentence is a palindrome, ignoring  
spaces, punctuation, and capitalization.
```

Task 4: ShoppingCart Class

PROMPT : Create a ShoppingCart class that allows adding items with their prices, removing items, and calculating the total cost.

The screenshot shows the Visual Studio Code interface with the following details:

- File Explorer:** Shows a folder named "2303AS1897_5" containing "2303AS1897_5.py" and "test_2303AS1897_5.py".
- Code Editor:** Displays the source code for "2303AS1897_5.py". The code defines a `ShoppingCart` class with methods for adding items, removing items, and calculating total cost. It also includes a `TestShoppingCart` class with several test cases.
- Terminal:** Shows the command line output of running the tests with Python 3.14.2. The terminal output indicates 13 tests ran in 0.002s and 0.003s respectively.
- Output:** Shows the results of the test runs.
- Problems:** Shows no problems.
- Status Bar:** Shows the current file is "2303AS1897_5.py", line 133, column 37, with 4 spaces, using UTF-8 encoding, and Python 3.14.2.

Task 5: Date Format Conversion

PROMPT : Write a function that converts a date from "YYYY-MM-DD" format to "DD-MM-YYYY" format.

The screenshot shows the Visual Studio Code interface with the following details:

- File Explorer:** Shows a folder named "2303AS1897_5" containing "2303AS1897_5.py" and "test_2303AS1897_5.py".
- Code Editor:** Displays the source code for "2303AS1897_5.py". The code defines a `convert_date_format` function that takes a date string and returns it in the "DD-MM-YYYY" format. It also includes a `TestDateConversion` class with several test cases.
- Terminal:** Shows the command line output of running the tests with Python 3.14.2. The terminal output indicates 13 tests ran in 0.002s and 0.003s respectively.
- Output:** Shows the results of the test runs.
- Problems:** Shows no problems.
- Status Bar:** Shows the current file is "2303AS1897_5.py", line 175, column 38, with 4 spaces, using UTF-8 encoding, and Python 3.14.2.

