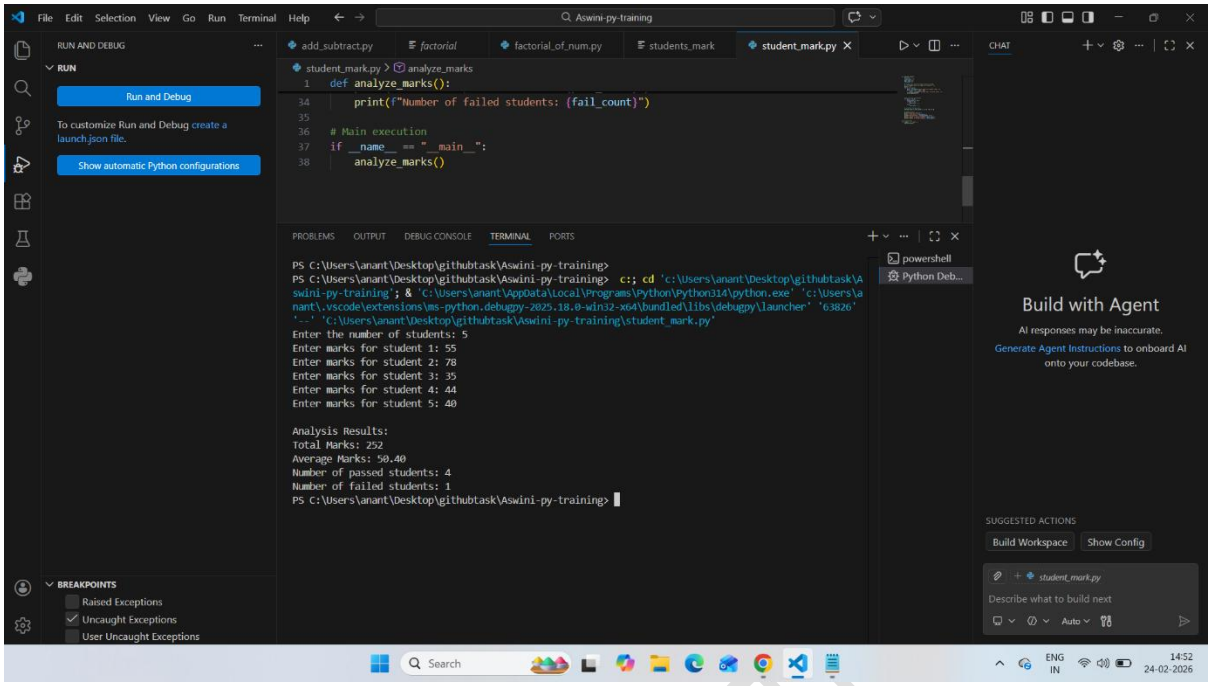
Mark_analysis

```
def analyze_marks():
    #initial
    total_marks = 0
    pass_count = 0
    fail_count = 0

    # student mark must be integer values in 0–100
    N = int(input("Enter the number of students: "))
    marks = []
    for i in range(N):
        mark = int(input(f"Enter marks for student {i+1}: "))
        while mark < 0 or mark > 100:
            mark = int(input("Invalid input. Re-enter marks: "))
        marks.append(mark)



    # pass or fail calculation
    for mark in marks:
        total_marks += mark
        if mark >= 40:
            pass_count += 1
        else:
```

```
            fail_count += 1

    # Calculate average marks
    average_marks = total_marks / N if N > 0 else 0

    # Output results
    print("\nAnalysis Results:")
    print(f"Total Marks: {total_marks}")
    print(f"Average Marks: {average_marks:.2f}")
    print(f"Number of passed students: {pass_count}")
    print(f"Number of failed students: {fail_count}")

# Main execution
if __name__ == "__main__":
    analyze_marks()
```



TIME COMPLEXICITY: O(n)

SPACE COMPLEXICITY: O(n)