AI-Assisted Problem Solving in Industry Scenarios

# NAME:J.ABHIRAM LAB TEST:3

# BATCH:06

# ROLL NO:2403A51342

## Question 1: Entertainment Sector – Data Structures with AI

Scenario: In the Entertainment sector, a company faces a challenge related to data

structures with ai.

Task: Use AI-assisted tools to solve a problem involving data structures with ai in this

context.

Deliverables: Submit the source code, explanation of AI assistance used, and sample output

### Prompt:

In the entertainment sector, use AI to analyze user movie ratings stored in a data structure (dictionary or list) and recommend top-rated movies. Use AI-assisted tools to help write efficient code for data handling, sorting, and recommendation.

### Code and Output:

### Observation:

The AI-assisted solution efficiently handled data using Python dictionaries and sorting. It demonstrated how AI can enhance data-driven decision-making and automate recommendation logic in entertainment applications.

## Question 2: Hospitality Sector – Backend API Development with AI

Scenario: In the Hospitality sector, a company faces a challenge related to backend api development.

Task: Use AI-assisted tools to solve a problem involving backend api development in this

context.

Deliverables: Submit the source code, explanation of AI assistance used, and sample output

## Prompt:

In the hospitality sector, develop a backend REST API using Flask that handles hotel booking operations such as viewing available rooms and adding new bookings. Use AI-assisted tools to generate efficient and clean backend code with proper route handling.

### Code:

### Output:

[  
 {"room\_id": 101, "type": "Deluxe", "available": true},  
 {"room\_id": 102, "type": "Suite", "available": true},  
 {"room\_id": 103, "type": "Standard", "available": false}  
]  
Request Body: {"room\_id": 101}  
Response: {"message": "Room 101 booked successfully!"}

### Observation:

The AI-assisted API effectively manages hotel booking data. It highlights how AI can streamline backend development by automating route setup, validation, and data handling. This leads to faster and cleaner API creation.