

## Mission Database: Manage the Field Agents (Python Version)

### STORY

You work for a secret military unit called Eagle Eye. You are building a tool to manage field agent data using Python and MySQL. Each agent has a code name, real name, current location, status, and number of missions completed.

### OBJECTIVES

Use MySQL script (SQL file) which contains the command to build the agents schema (given).

### DATABASE STRUCTURE

Database name: eagleEyeDB

Table: agents

Columns:

- id (INT, AUTO\_INCREMENT, PRIMARY KEY)
- codeName (VARCHAR)
- realName (VARCHAR)
- location (VARCHAR)
- status (VARCHAR) - values: "Active", "Injured", "Missing", "Retired"
- missionsCompleted (INT)

### Instructions:

1. Build a Python class to represent an agent (model)
  - a. It should implement the attributes
  - b. It should include the `__str__` function.
2. Create a Data Access Layer (DAL) class for database interaction
  - a. Handle MySql connection.
  - b. Implement CRUD operations for agents.
3. Write a main program to test the functionality
  - a. Add a well defined textual menu.

## PROJECT STRUCTURE

- models/
  - agent.py - defines the Agent class
- dal/
  - agent\_dal.py - handles MySQL operations with Agent data , handles database connection
- main.py - test program for adding, viewing, updating, and deleting agents

## CHECKLIST

- [x] MySQL DB and table created
- [x] Agent class with properties
- [x] DAL with all required methods
- [x] Parameterized queries used
- [x] Connection to MySQL tested
- [x] App tested with sample data