Project

Creating an Employee Database

Day 1

Write following functions as part of your project

1. **add\_employee**(emp\_id, emp\_name, emp\_DOJ, emp\_designation, emp\_salary)

Function adds the employee details in a file.   
Its similar to new onboarding of the employee. Take salary of the employee as default 0.

1. **add\_hr**(emp\_id, hr\_dept, hr\_role)

Function adds the hr details in another file.   
Its similar to new onboarding of a hr member.

***hr\_dept*** can be Recruitment, C&B, Admin. ***hr\_role*** can be Member, Lead, Manager.

1. **search\_employee**(emp\_id)

Take emp\_id as the input to the function. Read the employee file and print the details of the employee.

1. **search\_hr**(emp\_id)

Take emp\_id as the input to the function. Read the hr file and print the details of the hr.

**Optional**

1. **Upd\_employee**(emp\_id)

Take emp\_id as parameter to the function. Display current emp\_designation and emp\_salary and Prompt for the updated values. Update the record with the new values.

Day 2

Create a login.txt file with 1 record as below

admin admin

Create a script for **Login module**:

* Login module should prompt for the followings

Welcome to Employee System

Please Enter Login id: \_\_\_\_

Please Enter Password: \_\_\_\_\_

* Validate the login id and Password checking the login.txt file
  + If “Invalid”, display Error and come out
* If Valid,
  + If Login id is “admin” call **Admin Module**
  + If Login id is employee id of an employee call **Employee module** *(this will be explained tomorrow)*

Create a script for **Admin module**:

1. display menu of choices like below  
   Welcome Admin!!  
           Enter 1 to add employee  
           Enter 2 to remove employee  
           Enter 3 to add hr  
           Enter 4 to remove hr  
           Enter q to exit

Enter your Option 🡺

1. Option 1:
   * Prompt for emp\_id, emp\_name, emp\_DOJ, emp\_designation, emp\_salary
   * Add a record in “login.txt” with employee\_id and Firstname as password
   * Add a record in employee file through function already created in Day1
2. Option 2:
   * Prompt for emp\_id
   * Remove the record “login.txt” for the employee\_id
   * Remove the record from employee file through function already created in Day1
3. Option 3:
   * Prompt for emp\_id
   * As Hr is also an employee, first check if the emp\_id exist in employee file
   * If exist, Prompt for hr\_dept, hr\_role or else Display error and Quit
   * Add a record in hr file through function already created in Day1
4. Option 4:
   * Prompt for emp\_id
   * Remove the record from hr file through function already created Day1

**Optional**

* Update the above programs to generate the emp\_id automatically to avoid duplicates

Day 3

Create a function for **Employee**:

1. If Login\_id is NOT an HR, display menu of choices like below  
   Welcome <Employee\_name>!!  
           Enter 1 to view own details  
           Enter 2 to view all HR names  
           Enter q to exit

Enter your Option 🡺

* For Option 1, Just display all the employee details
* For Option 2, Display all the Department, HR names, and HR roles ***(Optional: sorted by Department)***

1. If Login\_id is an HR, display menu of choices like below

Welcome <Employee\_name> from HR!!  
        Enter 1 to view own details  
        Enter 2 to view All employees  
        Enter q to exit

Enter your Option 🡺

* For Option 1, Just display all the employee details *(Same as Option 1 above and hence should call the same function)*
* For Option 2, Prompt for Designation, and display details of all the employee with that designation
  + if designation is “All”, then display details of all the employees ***(Optional: sorted by Designation when “All”)***