Lab Dictionary Example

# Define a dictionary to store employee information

employee\_data = [

{"name": "John", "age": 30, "department": "Sales", "salary": 50000},

{"name": "Jane", "age": 25, "department": "Marketing", "salary": 60000},

{"name": "Mary", "age": 23, "department": "Marketing", "salary": 56000},

{"name": "Chloe", "age": 35, "department": "Engineering", "salary": 70000},

{"name": "Mike", "age": 32, "department": "Engineering", "salary": 65000},

{"name": "Peter", "age": 40, "department": "Sales", "salary": 60000}

]

def get\_employees\_by\_age\_range(age\_lower\_limit, age\_upper\_limit):

result = []

# check for age limits and append the item to result

for item in employee\_data:

if int(item["age"]) > int(age\_lower\_limit) and int(item["age"]) < int(age\_upper\_limit):

result.append(item)

return result

def calculate\_average\_salary():

total = 0

average = 0

salary=[]

for item in employee\_data:

total = 0

person =0

if item["salary"]:

value = item["salary"]

total = total + value

person = person + 1

average = total / person

return average

def get\_employees\_by\_dept(department):

result = []

for item in employee\_data:

if item["department"]==department:

result.append(item)

return result

def display\_all\_records():

print(("Name" + "\t" +"Age" +"\t" +"Department" +"\t" +"Salary" ).expandtabs(15))

for item in employee\_data:

print((item["name"] + "\t" + str(item["age"]) + "\t" + item["department"] + "\t" + str(item["salary"])).expandtabs(15))

def display\_records(employee\_info):

print(("Name" + "\t" +"Age" +"\t" +"Department" +"\t" +"Salary" ).expandtabs(15))

for item in employee\_info:

print((item["name"] + "\t" + str(item["age"]) + "\t" + item["department"] + "\t" + str(item["salary"])).expandtabs(15))

def display\_main\_menu():

print("\n----- Employee information Tracker -----")

print("Select option\n")

print("1 - Display all records")

print("2 - Display average salary")

print("3 - Display employee within age range")

print("4 - Display employee in a department")

print("Q - Quit")

option = input("Enter selection =>")

if option == '1':

display\_all\_records()

elif option == '2':

average\_salary = calculate\_average\_salary()

print("Average salary = " + str(average\_salary))

elif option == '3':

age\_lower\_limit = input("age (Lower Limit) = ")

age\_upper\_limit = input("age (Uper Limit) = ")

employee\_info = get\_employees\_by\_age\_range(age\_lower\_limit, age\_upper\_limit)

display\_records(employee\_info)

elif option == '4':

department = input("Name of Department = ")

employee\_info = get\_employees\_by\_dept(department)

display\_records(employee\_info)

elif option == 'Q':

quit()

def main():

while (True):

display\_main\_menu()

if \_\_name\_\_ == "\_\_main\_\_":

main()

Pytest

import pytest

from employee\_info import get\_employees\_by\_age\_range

def test\_get\_employees\_by\_age\_range():

expected\_results = [

{"name": "John", "age": 30, "department": "Sales", "salary": 50000},

{"name": "Chloe", "age": 35, "department": "Engineering", "salary": 70000},

{"name": "Mike", "age": 32, "department": "Engineering", "salary": 65000}]

given\_result = get\_employees\_by\_age\_range(30,35)

if given\_result==expected\_results:

assert(True)

from employee\_info import calculate\_average\_salary

def test\_calculate\_average\_salary():

expected\_results = 60000

given\_result = calculate\_average\_salary()

if given\_result == expected\_results:

assert(True)

from employee\_info import get\_employees\_by\_dept

def test\_get\_employees\_by\_dept():

expected\_results= [

{"name": "Jane", "age": 25, "department": "Marketing", "salary": 60000},

{"name": "Mary", "age": 23, "department": "Marketing", "salary": 56000}

]

given\_result= get\_employees\_by\_dept("Marketing")

if given\_result==expected\_results:

assert(True)