#

class Employee:

def \_\_init\_\_(self, emp\_name, emp\_id, emp\_salary, emp\_department):

self.emp\_name = emp\_name

self.emp\_id = emp\_id

self.emp\_salary = emp\_salary

self.emp\_department = emp\_department

def assign\_department(self, new\_department):

self.emp\_department = new\_department

def calculate\_emp\_salary(self, hours\_worked):

if hours\_worked > 50:

overtime\_hours = hours\_worked - 50

overtime\_amount = overtime\_hours \* (self.emp\_salary / 50)

total\_salary = self.emp\_salary + overtime\_amount

else:

total\_salary = self.emp\_salary

return total\_salary

def print\_employee\_details(self):

print(f"Employee ID: {self.emp\_id}")

print(f"Name: {self.emp\_name}")

print(f"Salary: {self.emp\_salary}")

print(f"Department: {self.emp\_department}")

employeez = [

Employee("Radha", "E7876", 100000, "Developer"),

Employee("Shiny", "E7499", 45000, "Tester"),

Employee("David", "E7900", 50000, "Analyst"),

Employee("Raja", "E7698", 55000, "Designer")

]

for emp in employeez:

emp.print\_employee\_details()

total\_salary = emp.calculate\_emp\_salary(60)

print(f"Total Salary for 60 hours: {total\_salary:.2f}\n")

employeez[0].assign\_department("Senior Developer")

print("After changing department:")

employeez[0].print\_employee\_details()