def format\_salary(salary):

"""Helper to format salary with Rs. and commas."""

return f"Rs. {int(salary):,}"

try:

employees = []

with open("employees.txt", "r") as file:

for line in file:

try:

emp\_id, name, dept, salary = line.strip().split(",")

employees.append({

"id": emp\_id.strip(),

"name": name.strip(),

"dept": dept.strip(),

"salary": float(salary.strip())

})

except ValueError:

# Skips malformed lines

print(f"Skipping invalid line: {line.strip()}")

# Filter: IT or HR with salary > 50,000

filtered = list(filter(

lambda e: e["dept"] in ["IT", "HR"] and e["salary"] > 50000,

employees

))

# Map: keep only id, name, formatted salary

mapped = list(map(

lambda e: {"id": e["id"], "name": e["name"], "salary": format\_salary(e["salary"])},

filtered

))

# Write results into new file

with open("filtered\_employees.txt", "w") as f:

for emp in mapped:

f.write(f"{emp['id']}, {emp['name']}, {emp['salary']}\n")

print("Filtered employees written to filtered\_employees.txt")

except FileNotFoundError:

print("Error: employees.txt not found!")