Name:

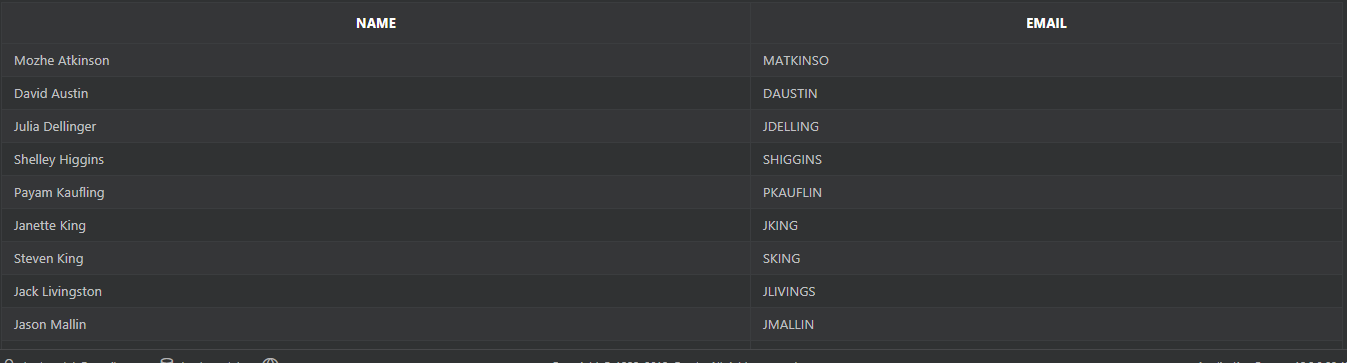
Exam Set: ODD or EVEN

# Problem 1

Create a list of every employee's first name concatenated to a space and the employee's last name, and the email of all employees where the email address contains the string 'IN'.

**Tables Used: OEHR\_Employees**

**Output:**



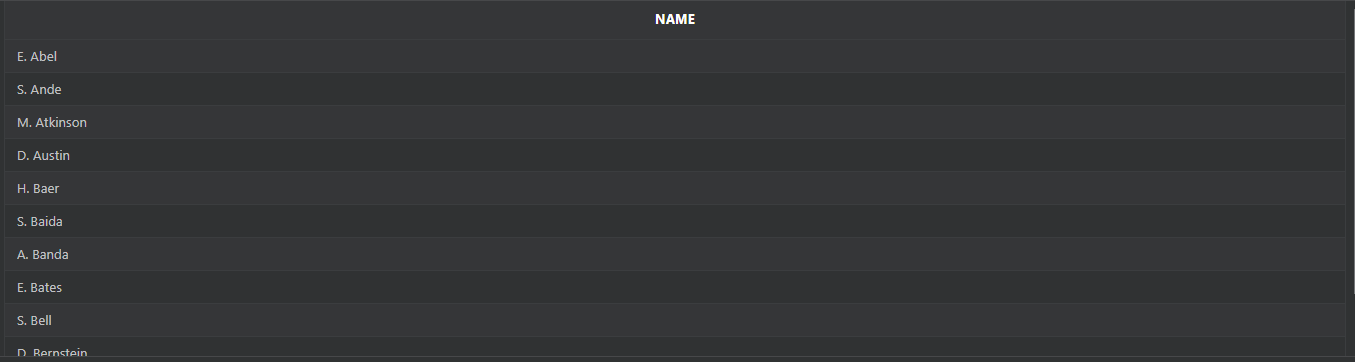
**Answer:**

# Problem 2

Create a list that includes the first initial of every employee's first name, a space, and the last name of the employee.

**Tables Used: OEHR\_Employees**

**Output**



**Answer:**

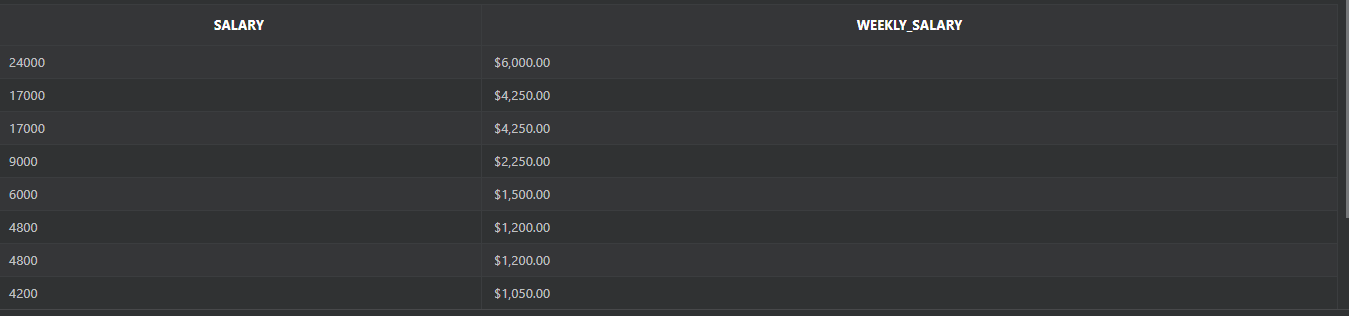
# Problem 3

Create a list of weekly salaries from the employees table where the weekly salary is between 700 and 3000.

The salaries should be formatted to include a $-sign and have two decimal points like: $9999.99.

**Tables Used: OEHR\_Employees**

**Output:**



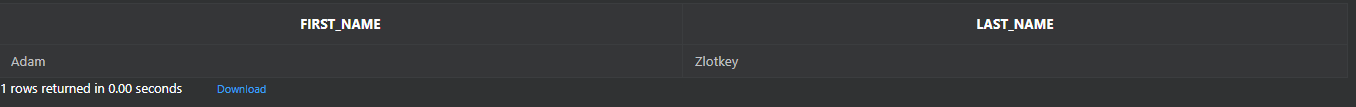
**Answer:**

# Problem 4

Create a list of 'smallest' last name and the 'highest' last name from the employees table.

**Tables Used: OEHR\_Employees**

**Output:**



**Answer:**

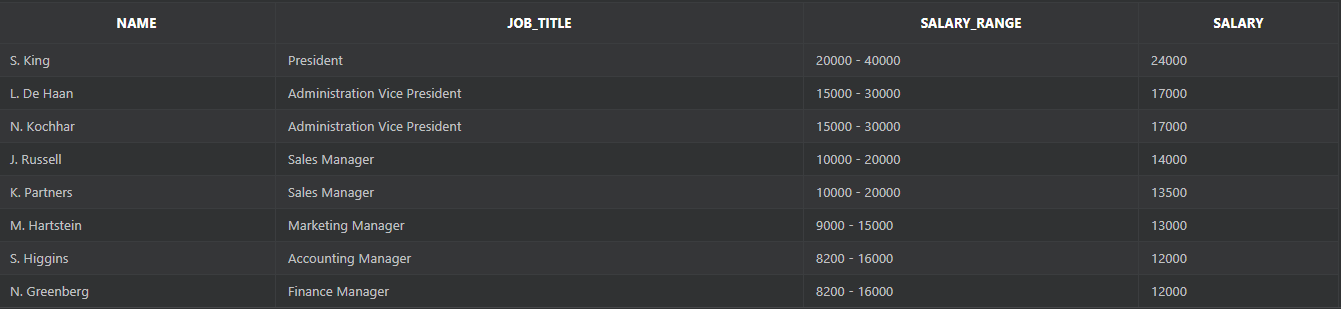
# Problem 5

Create a list of every employee's job, the salary ranges within the job, and the employee's salary.

List the lowest and highest salary range within each job with a dash to separate the salaries like this: 100 – 200.

**Tables Used: OEHR\_Employees, OEHR\_Jobs**

**Output:**



**Answer:**

# Problem 6

Create a list of every employee and his related job title sorted by job\_title.

**Tables Used: OEHR\_Employees, OEHR\_Jobs**

**Output:**



**Answer:**

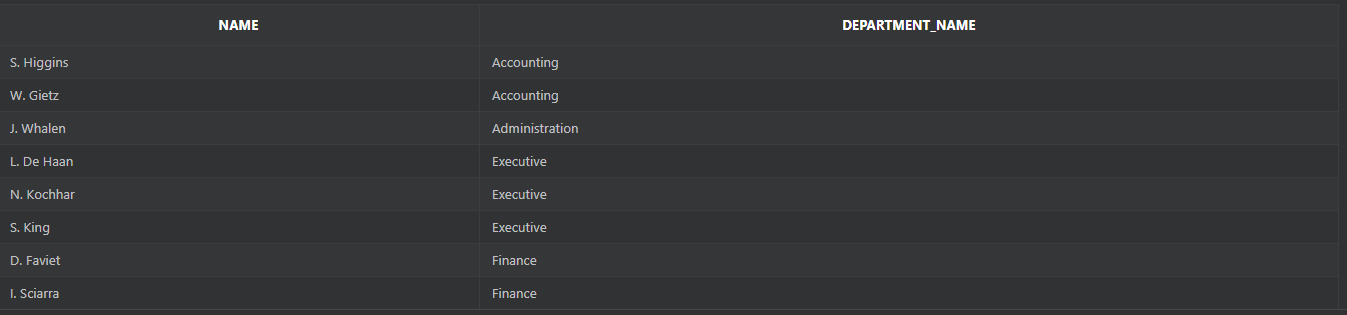
# Problem 7

Using an ANSII join method, create a list of every employee's first initial and last name, and department name.

Make sure the tables are joined on all of the foreign keys declared between the two tables.

**Tables Used: OEHR\_Employees, OEHR\_Departments**

**Output:**



**Answer:**

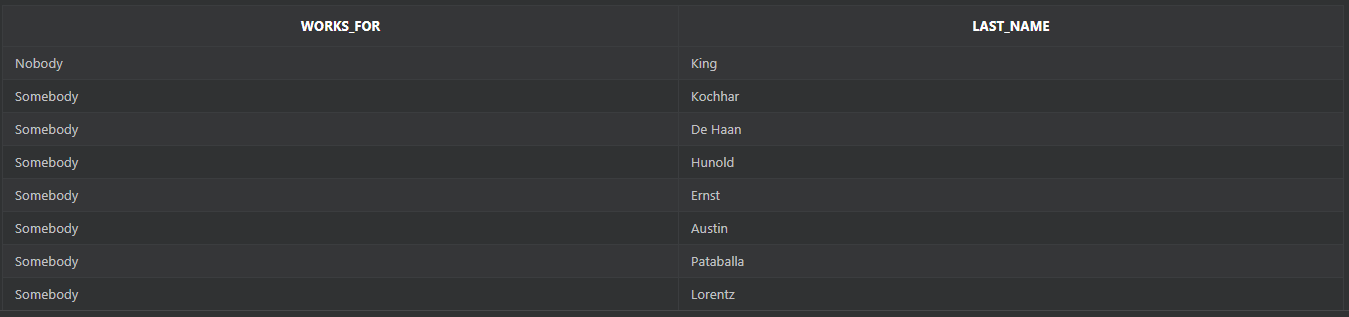
# Problem 8

Create a list of every employee's last name, and the word nobody or somebody depending on whether or not the employee has a manager.

Use the Oracle DECODE function to create the list.

**Tables Used: OEHR\_Employees**

**Output:**



**Answer:**

# Problem 9

Create a list of every employee's first initial and last name, salary, and a yes or no to show whether or not an employee makes a commission.

Use the Oracle DECODE function to create the list.

**Tables Used: OEHR\_Employees**

**Output:**



**Answer:**

# Problem 10

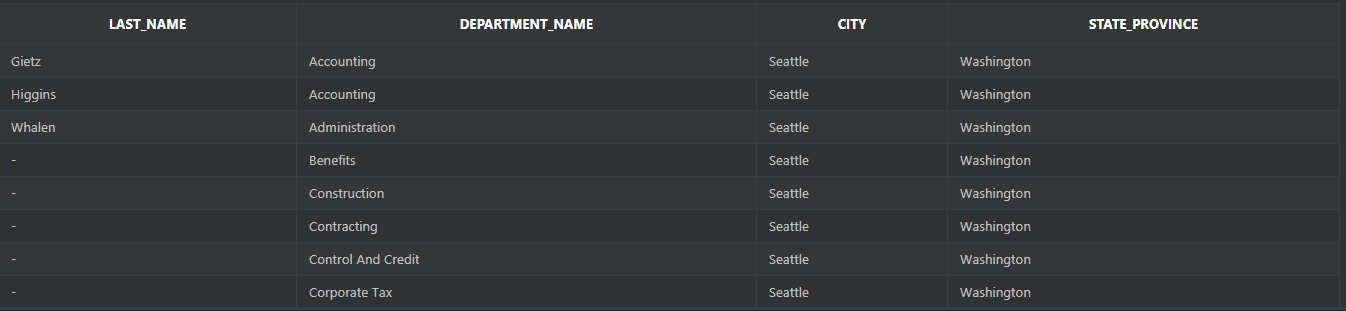
Create a list of every employee's last name, department name, city, and state\_province.

Include departments without employees.

An outer join is required.

**Tables Used: OEHR\_Employees, OEHR\_Departments, OEHR\_Locations**

**Output:**



**Answer:**

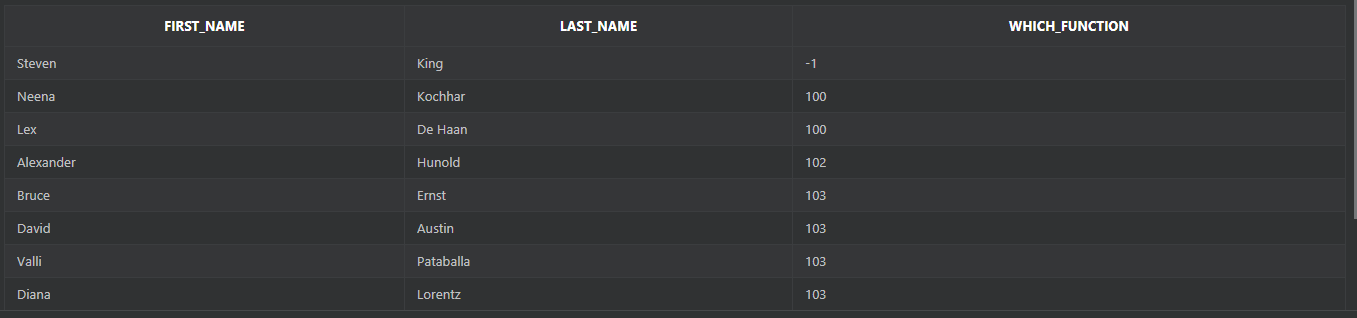
# Problem 11

Create a list of every employee's first and last names, and the first occurrence of: commission\_pct, manager\_id, or -1.

If an employee gets commission, display the commission\_pct column; if no commission, then display his manager\_id; if he has neither commission nor manager, then the number -1.

**Tables Used: OEHR\_Employees**

**Output:**



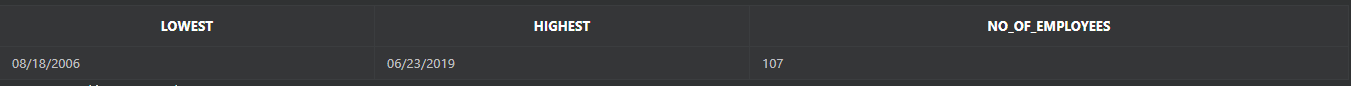
**Answer:**

# Problem 12

Produce a list of the earliest hire date, the latest hire date, and the number of employees from the employees table.

**Tables Used: OEHR\_Employees**

**Output:**



**Answer:**

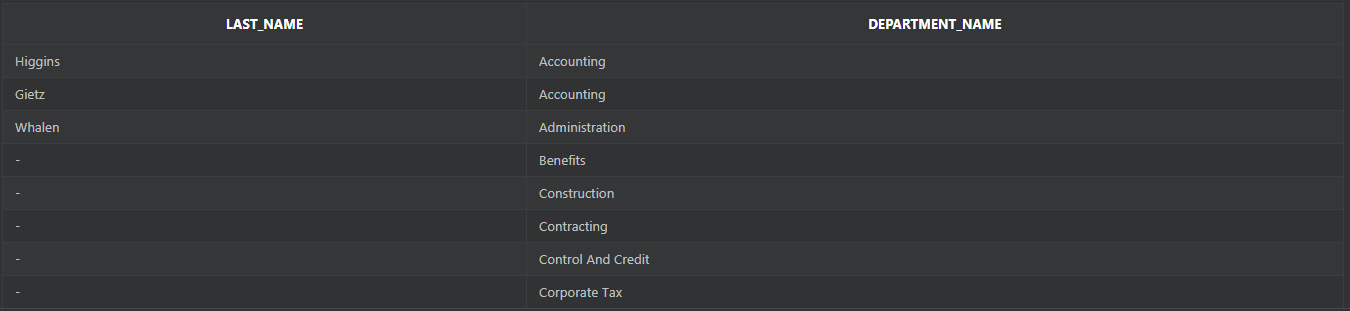
# Problem 13

Produce a list of every employee's last name and department name.

Include both employees without departments, and departments without employees.

**Tables Used: OEHR\_Employees, OEHR\_Departments**

**Output:**



**Answer:**

# Problem 14

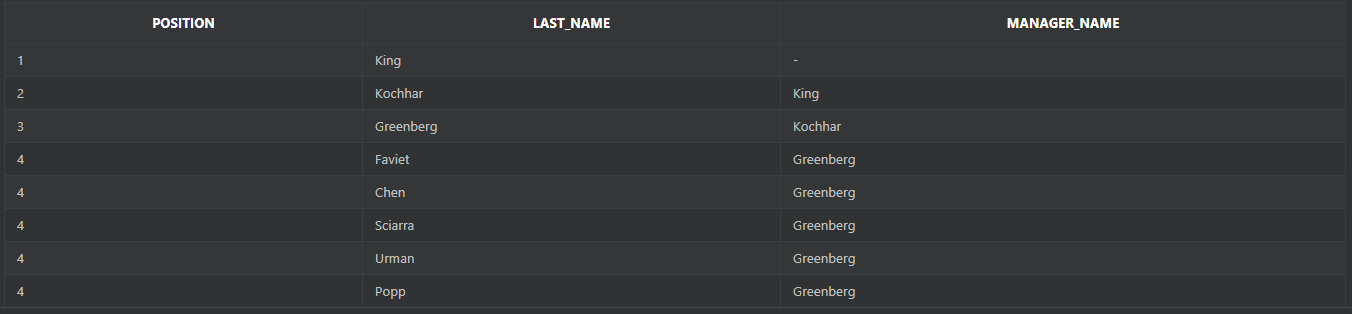
Create a treewalking list of every employee's last name, his manager's last name, and his position in the company.

The top level manager has position 1, this manager's subordinates position 2, their subordinates position 3, and so on.

Start the listing with employee number 100.

**Tables Used: OEHR\_Employees**

**Output:**



**Answer:**

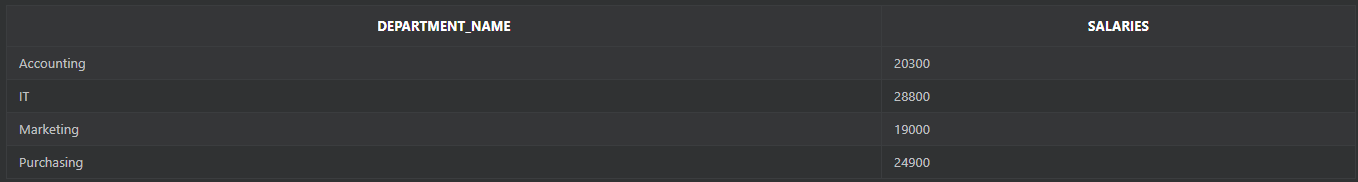
# Problem 15

Create a list of department names and the departmental costs (salaries added up).

Include only departments whose salary costs are between 15000 and 31000, and sort the listing by the cost.

**Tables Used: OEHR\_Employees, OEHR\_Departments**

**Output:**



**Answer:**

# Problem 16

Create a list of department names and their monthly costs (salaries added up).

**Tables Used: OEHR\_Employees, OEHR\_Departments**

**Output:**



**Answer:**

# Problem 17

Create a list of department names, the manager id, manager name (employee last name) of that department, and the average salary in each department.

**Tables Used: OEHR\_Employees, OEHR\_Departments**

**Output:**



**Answer:**

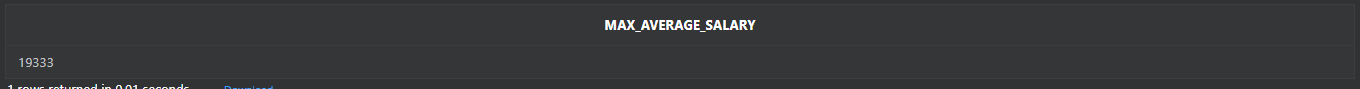
# Problem 18

Show the highest average salary for the departments in the employees table.

Round the result to the nearest whole number.

**Tables Used: OEHR\_Employees**

**Output:**



**Answer:**

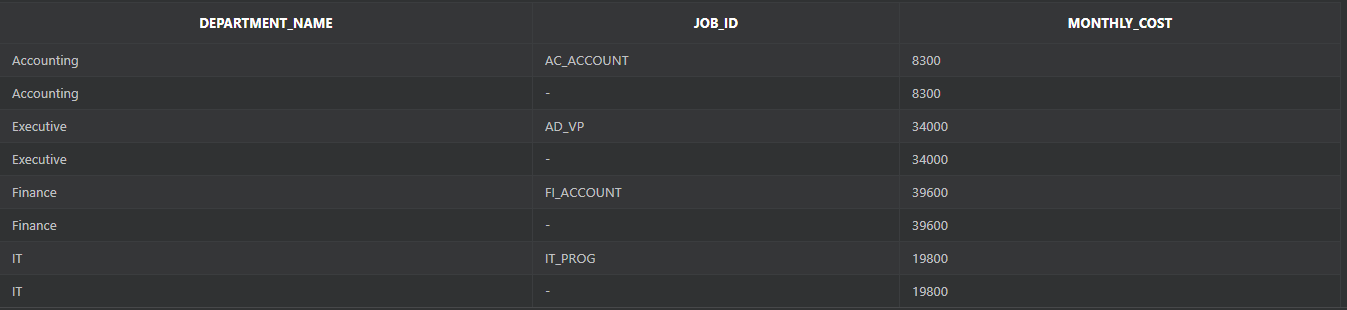
# Problem 19

Create a list of department names, and job\_ids.

Calculate the monthly salary cost for each job\_id within a department, for each department, and for all departments added together.

**Tables Used: OEHR\_Employees, OEHR\_Departments**

**Output:**



**Answer:**

# Problem 20

Create a list of department names, and job\_ids.

Calculate the monthly salary cost for each job\_id within a department, for each department, for each group of job\_ids irrespective of the department, and for all departments added together. (Hint: Cube)

**Tables Used: OEHR\_Employees, OEHR\_Departments**

**Output:**



**Answer:**

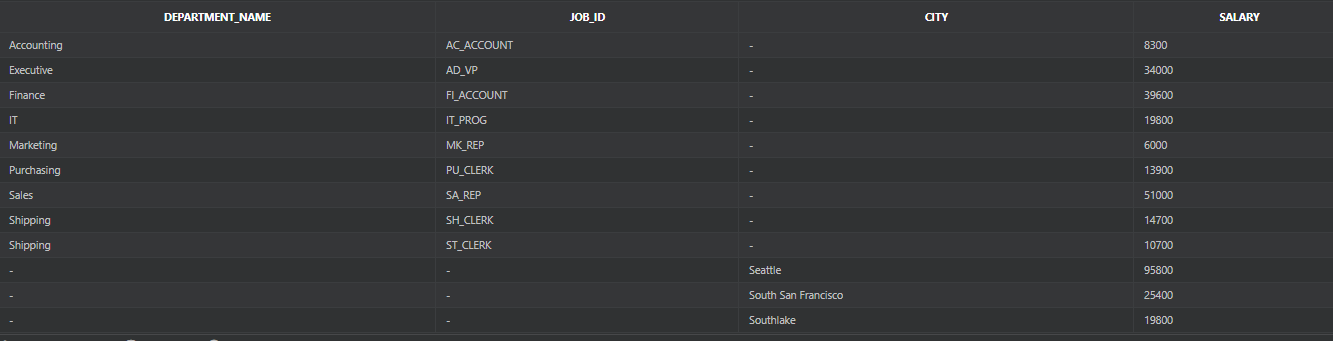
# Problem 21

Create a list that includes the monthly salary costs for each job title within a department.

In the same list, display the monthly salary cost per city. (Hint: Grouping Sets)

**Tables Used: OEHR\_Employees, OEHR\_Departments, OEHR\_Locations**

**Output:**



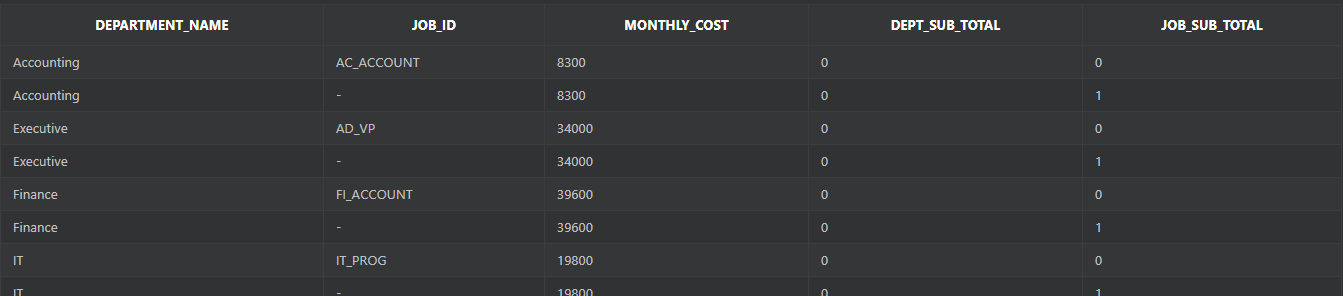
**Answer:**

# Problem 22

Expand the previous list to also show if the department\_id or job\_id was used to create the subtotals shown in the output. (Hint: Cube, Grouping)

**Tables Used: OEHR\_Employees, OEHR\_Departments**

**Output:**



**Answer:**

# Problem 23

Create a list of each employee's first initial and last name, salary, and department name for each employee earning more than the average for his department.

**Tables Used: OEHR\_Departments, OEHR\_Employees**

**Output:**



**Answer:**

# Problem 24

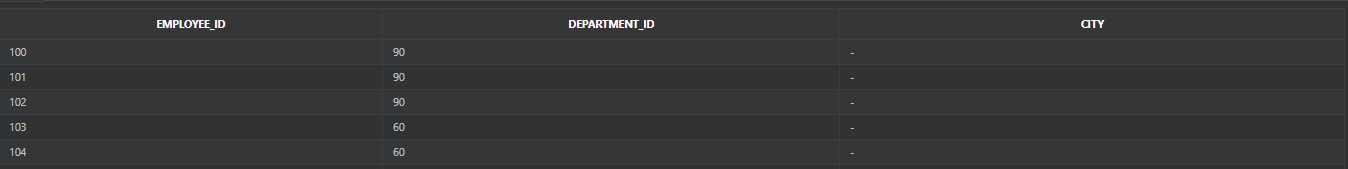
Create a list of employee names as shown and department ids.

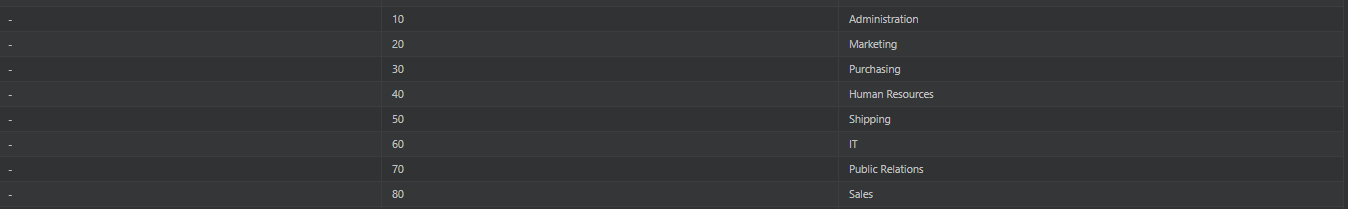
In the same report, list the department ids and department names. And finally, list the cities.

The rows should not be joined, just listed in the same report. (Hint: Union)

**Tables Used: OEHR\_Employees, OEHR\_Departments, OEHR\_Locations**

**Output:**





**Answer:**