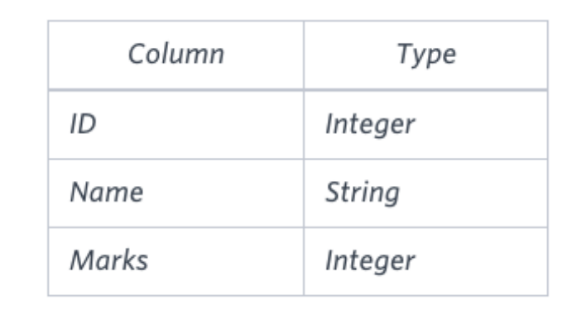
**Q1**

Query the *Name* of any student in **STUDENTS** who scored higher than  *Marks*. Order your output by the *last three characters* of each name. If two or more students both have names ending in the same last three characters (i.e.: Bobby, Robby, etc.), secondary sort them by ascending *ID*.

**Input Format**

The **STUDENTS** table is described as follows:



**SOLUTION**: SELECT NAME AS NAME FROM STUDENTS

WHERE MARKS > 75

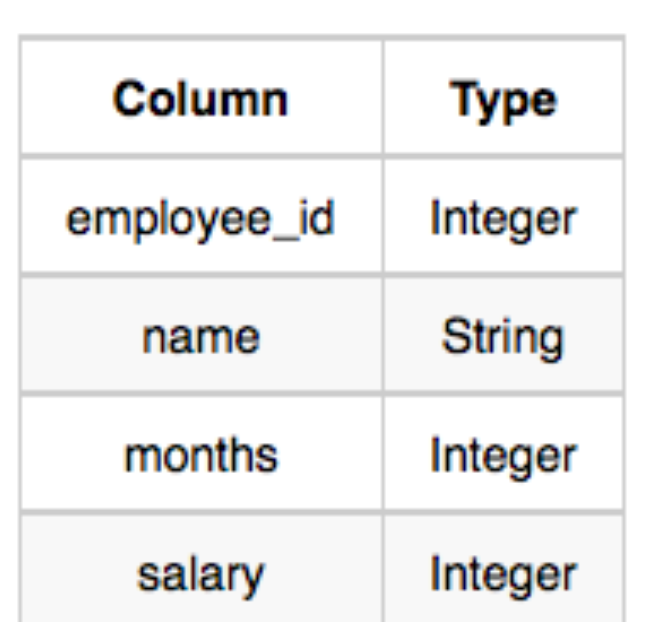
ORDER BY SUBSTR(NAME, LENGTH(NAME)-2, 3), ID;

**Q2**

Write a query that prints a list of employee names (i.e.: the *name* attribute) from the **Employee** table in alphabetical order.

**Input Format**

The **Employee** table containing employee data for a company is described as follows:



**SOLUTION :** SELECT NAME FROM EMPLOYEE

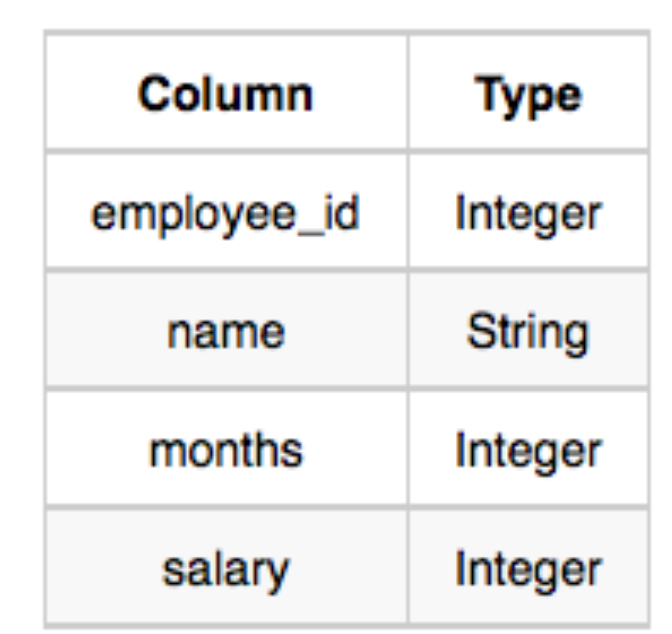
ORDER BY NAME ASC;

**Q3**

Write a query that prints a list of employee names (i.e.: the *name* attribute) for employees in **Employee** having a salary greater than  per month who have been employees for less than  months. Sort your result by ascending *employee\_id*.

**Input Format**

The **Employee** table containing employee data for a company is described as follows:



**SOLUTION** : SELECT NAME FROM EMPLOYEE

WHERE SALARY > 2000 AND MONTHS < 10

ORDER BY EMPLOYEE\_ID ASC;