**DATABASE CONCEPTS**

**What is Database?**

A database is an organized collection of data. It is the collection of schemas, tables, queries, reports, views, and other objects.

**What is Table?**

A table is a set of data elements (values) using a model of vertical columns (identifiable by name) and horizontal rows, the cell being the unit where a row and column intersect. A table has a specified number of columns, but can have any number of rows.

**What are rows and columns?**

Row also called a record or tuple—represents a single, implicitly structured data item in a table. In simple terms, a database table can be thought of as consisting of rows and columns or fields.

**Example for Inner Join?**

SQL> desc employees;

Name Null? Type

----------------------------------------- -------- ----------------------------

EMPLOYEE\_ID NOT NULL NUMBER(6)

FIRST\_NAME VARCHAR2(20)

LAST\_NAME NOT NULL VARCHAR2(25)

EMAIL NOT NULL VARCHAR2(25)

PHONE\_NUMBER VARCHAR2(20)

HIRE\_DATE NOT NULL DATE

JOB\_ID NOT NULL VARCHAR2(10)

SALARY NUMBER(8,2)

COMMISSION\_PCT NUMBER(2,2)

MANAGER\_ID NUMBER(6)

DEPARTMENT\_ID NUMBER(4)

SQL> desc departments;

Name Null? Type

----------------------------------------- -------- ----------------------------

DEPARTMENT\_ID NOT NULL NUMBER(4)

DEPARTMENT\_NAME NOT NULL VARCHAR2(30)

MANAGER\_ID NUMBER(6)

LOCATION\_ID NUMBER(4)

SQL> select e.first\_name,e.salary,e.job\_id,d.\*

2 from employees e inner join departments d

3 on e.department\_id=d.department\_id

4 where job\_id='ST\_CLERK';

FIRST\_NAME SALARY JOB\_ID DEPARTMENT\_ID

-------------------- ---------- ---------- -------------

DEPARTMENT\_NAME MANAGER\_ID LOCATION\_ID

------------------------------ ---------- -----------

Julia 3200 ST\_CLERK 50

Shipping 121 1500

Irene 2700 ST\_CLERK 50

Shipping 121 1500

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| EMPLOYEE TABLE |  |  |  |  |
|  |  |  |  |  |
|  | EID | ENAME | SALARY | DNAME |
|  | 1 | A | 100 | DA |
|  | 2 | B | 200 | DB |
|  | 3 | C | 300 | DC |
|  | 4 | D | 400 |  |
|  | 5 | E | 500 | DE |
|  |  |  |  |  |
| DEPARTMENT TABLE |  |  |  |  |
|  |  |  |  |  |
|  | DID | DNAME |  |  |
|  | D1 | DA |  |  |
|  | D2 | DB |  |  |
|  | D3 | DC |  |  |
|  | D4 |  |  |  |
|  | D5 | DE |  |  |
|  |  |  |  |  |

**Example for Left Outer Join?**

Select e.ename,e.salary,d.\*

from employee e LEFT JOIN department d

on e.dname=d.dname;

o/p:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| ENAME | SALARY | DNAME | DID | DNAME |
| A | 100 | DA | D1 | DA |
| B | 200 | DB | D2 | DB |
| C | 300 | DC | D3 | DC |
| E | 500 | DE | D5 | DE |
| D | 400 |  |  |  |

**Example for Right Outer Join?**

Select e.ename,e.salary,d.\*

from employee e RIGHT JOIN department d

on e.dname=d.dname;

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| ENAME | SALARY | DNAME | DID | DNAME |
| A | 100 | DA | D1 | DA |
| B | 200 | DB | D2 | DB |
| C | 300 | DC | D3 | DC |
| E | 500 | DE | D5 | DE |
|  |  |  | D4 |  |

**Example for Full Outer Join?**

Select e.ename,e.salary,d.\*

from employee e FULL JOIN department d

on e.dname=d.dname;

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| ENAME | SALARY | DNAME | DID | DNAME |
| A | 100 | DA | D1 | DA |
| B | 200 | DB | D2 | DB |
| C | 300 | DC | D3 | DC |
| D | 400 |  | D4 |  |
| E | 500 | DE | D5 | DE |

**Example for Max, Sum, Avg?**

SQL> select max(salary) from employees;

MAX(SALARY)

-----------

24000

SQL> select sum(salary) from employees;

SUM(SALARY)

-----------

691400

SQL> select avg(salary) from employees;

AVG(SALARY)

-----------

6461.68224

**Example for Where condition?**

SQL> select \* from employees where job\_id='AC\_MGR' order by department\_id;

EMPLOYEE\_ID FIRST\_NAME LAST\_NAME EMAIL

----------- -------------------- ------------------------- -------------------------

PHONE\_NUMBER HIRE\_DATE JOB\_ID SALARY COMMISSION\_PCT MANAGER\_ID DEPARTMENT\_ID

-------------------- --------- ---------- ---------- -------------- ---------- -------------

205 Shelley Higgins SHIGGINS

515.123.8080 07-JUN-94 AC\_MGR 12000 101 110

**Example for group by and having?**

SQL> select first\_name,sum(salary) from emp\_new group by first\_name having sum(salary)>10000;

FIRST\_NAME SUM(SALARY)

-------------------- -----------

Steven 26200

Nancy 12000

Michael 15900

Peter 21500

Alberto 12000

Ellen 11000

Shelley 12000

John 24900

William 15700

Karen 16000

Eleni 10500

FIRST\_NAME SUM(SALARY)

-------------------- -----------

Clara 10500

Lex 17000

Gerald 11000

Den 11000

Neena 17000

David 21100

Lisa 11500

Alexander 12100

19 rows selected.

**Example for Primary Key and Foreign Key**

/\*Creating Parent table\*/

Create table dept(

DeptID int PRIMARY KEY, /\*defining primary key\*/

Dname varchar2(20) NOT NULL,

Daddr varchar2(20) NULL);

1 table created

/\*Creating Child table\*/

Create table Emp(

EmpID int PRIMARY KEY, /\*defining primary key\*/

Ename varchar2(20) NOT NULL,

Esal int NULL,

DeptID int FOREIGN KEY REFERENCES dept(DeptID) /\*defining foreign key\*/

);

1 table created

**Finding second highest salary**

SQL> select max(salary) from emp\_new where salary<(select max(salary) from emp\_new);

MAX(SALARY)

-----------

17000