CHAPTER – 2

SQL – STRUCTURED QUERY LANGUAGE

SQL – Structured Query Language.
(SEQUEL – Simple English Query Language.)

SQL – it is a language to talk to the database/to access the database.
SQL – it is a domain specific language (RDBMS).
SQL – it is a declarative (what to do) language.

Using SQL, you can write data in to a RDBMS and read data from a RDBMS. You can also create, modify and delete data in a DB.

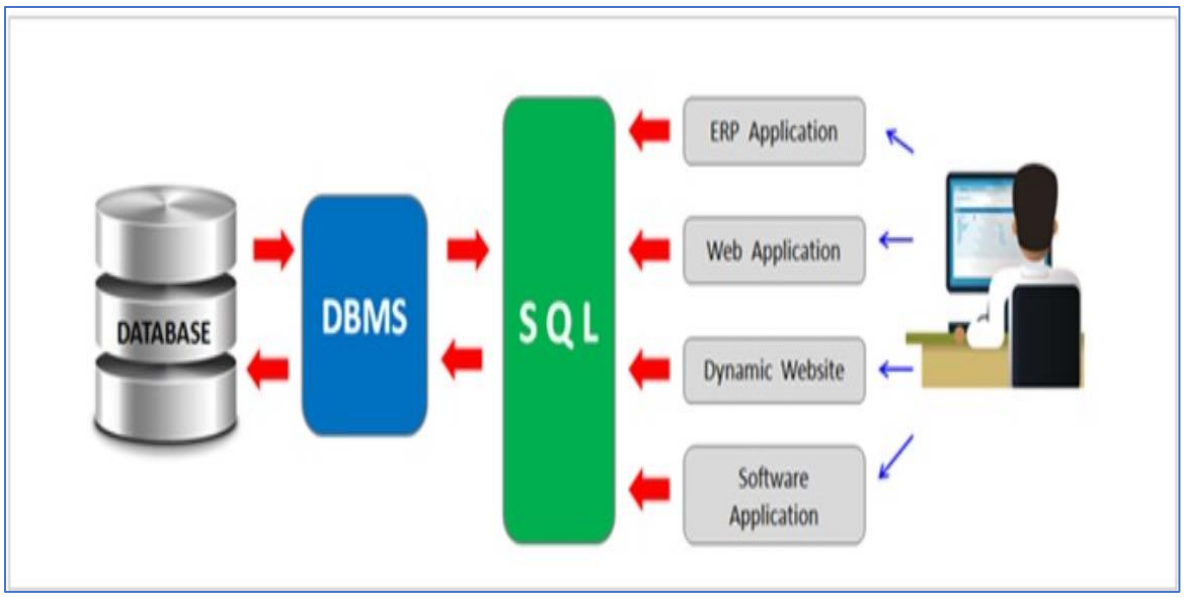
Example: To find the student's name who belongs to Bengaluru?
*SELECT * FROM STUDENT WHERE ADDRESS = 'BENGALURU';*

Almost all of the RDBMS such as Oracle, MySQL, Microsoft SQL Server, PostgreSQL, IBM DB2, SQLite, etc... uses SQL to manage their data. Different RDBMS have created their own version of SQL but majority of the functionality and the syntax are common across all the DB.

** **PL/SQL** – is a procedural language (here what to do & how to do) designed specifically to embrace SQL statements within its syntax.

SQL – it is a language, whereas SQL server is a database.

To work on SQL, a DB software (RDBMS) is required. SQL is not case sensitive.



The DB user interacts with the application program. The application programs communicate with the DBMS using SQL (Structured Query Language) commands. The DBMS effectively functions as an interface between the application program and database. The DBMS operates on the database as per the SQL Commands.

Using SQL Commands, we perform **CRUD operations** like:

CREATE – Create the Database/Tables
RETREIVE - access the data from DB
UPDATE – Update the Data and
DELETE – Remove the data from the table.

Using SQL commands we can perform the below operations, by writing SQL queries.

CREATE Database
CREATE Table
ALTER Database
ALTER Table
SELECT(READ) Data
UPDATE Data
DELETE Database
DELETE Table
DELETE Data

SQL Statements/Commands: DDL, DML, DCL, TCL, DQL

Select is a DQL command and it is used to retrieve and view data from one or many tables in a DB.

Few examples:

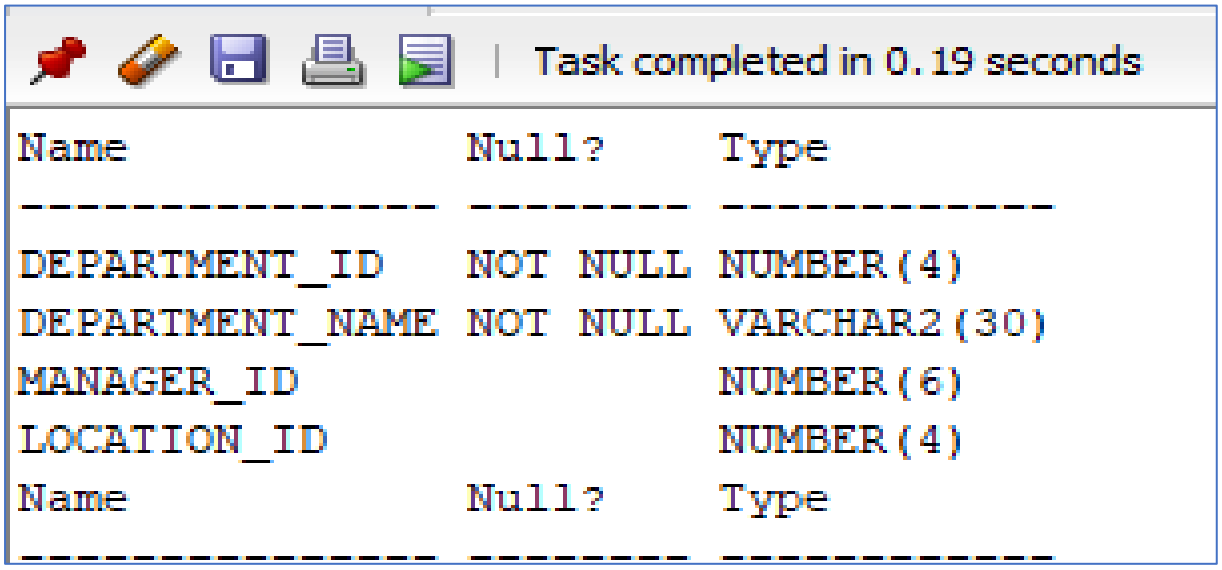
SQL > SELECT * FROM TAB;

SQL | All Rows Fetched: 32 in 0.035 seconds

	TNAME	TABTYPE	CLUSTERID
1	BONUS	TABLE	(null)
2	COUNTRIES	TABLE	(null)
3	DEPARTMENTS	TABLE	(null)
4	EMPLOYEES	TABLE	(null)
5	EMPLOYEES11	TABLE	(null)
6	EMPLOYEES_V1	VIEW	(null)
7	EMP_DETAILS_VIEW	VIEW	(null)
8	JOBS	TABLE	(null)
9	JOB_HISTORY	TABLE	(null)
10	LOCATIONS	TABLE	(null)

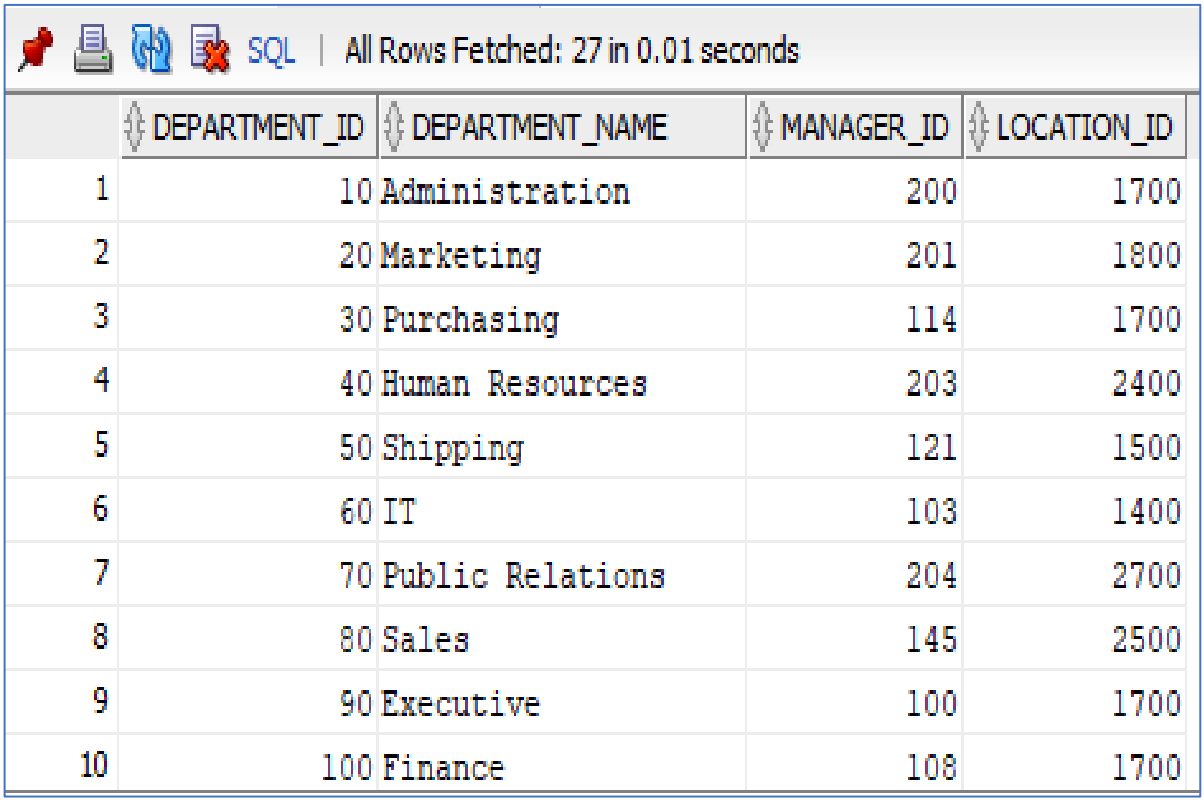
This query gives the list of tables.
Note: * means → selects all

SQL > DESC DEPARTMENTS;



This query gives the description of the table "department".
The description of the table has **column names, constraints, datatypes.**

SQL > SELECT * FROM DEPARTMENTS;



	DEPARTMENT_ID	DEPARTMENT_NAME	MANAGER_ID	LOCATION_ID
1	10	Administration	200	1700
2	20	Marketing	201	1800
3	30	Purchasing	114	1700
4	40	Human Resources	203	2400
5	50	Shipping	121	1500
6	60	IT	103	1400
7	70	Public Relations	204	2700
8	80	Sales	145	2500
9	90	Executive	100	1700
10	100	Finance	108	1700

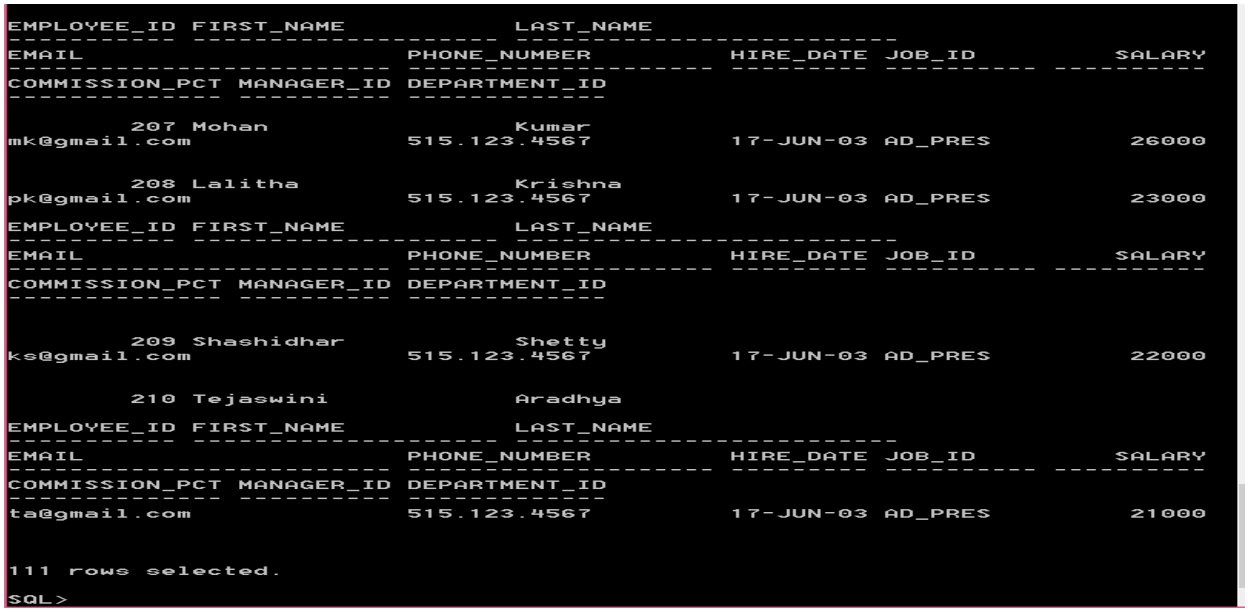
This query gives all the records of the table "departments".

SQL > SELECT * FROM EMPLOYEES;

This query gives all the records of the table "employees".

In SQL PLUS utility,

SQL > SELECT * FROM EMPLOYEES;



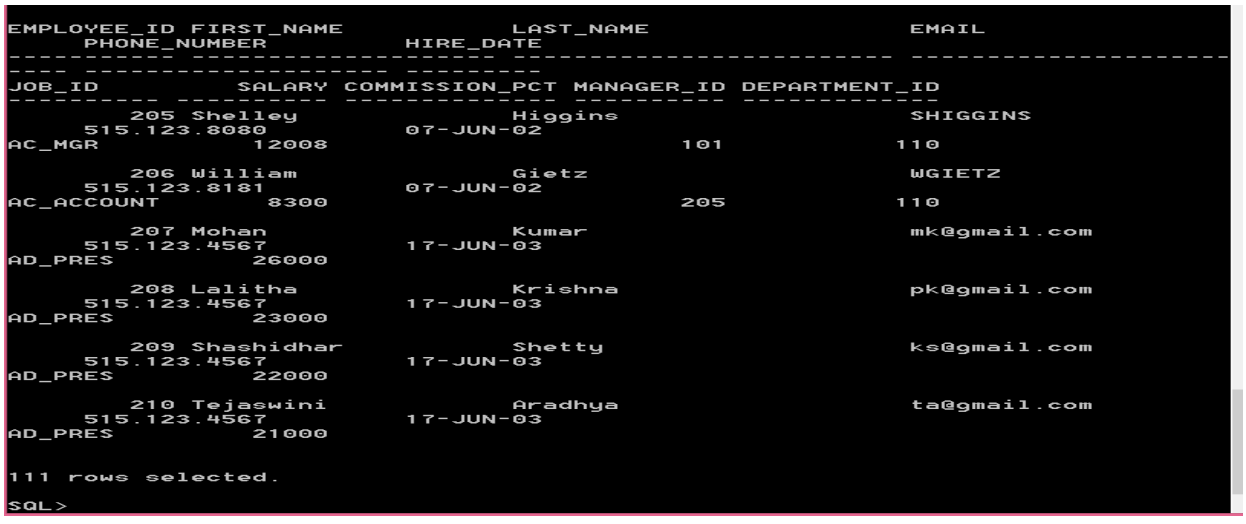
```
EMPLOYEE_ID FIRST_NAME            LAST_NAME
----------- --------------------- -------------------------
EMAIL                     PHONE_NUMBER         HIRE_DATE JOB_ID          SALARY
------------------------- -------------------- --------- ---------- ----------
COMMISSION_PCT MANAGER_ID DEPARTMENT_ID
-------------- ---------- -------------

        207 Mohan                 Kumar
mk@gmail.com              515.123.4567         17-JUN-03 AD_PRES          26000

        208 Lalitha               Krishna
pk@gmail.com              515.123.4567         17-JUN-03 AD_PRES          23000
EMPLOYEE_ID FIRST_NAME            LAST_NAME
----------- --------------------- -------------------------
EMAIL                     PHONE_NUMBER         HIRE_DATE JOB_ID          SALARY
------------------------- -------------------- --------- ---------- ----------
COMMISSION_PCT MANAGER_ID DEPARTMENT_ID
-------------- ---------- -------------

        209 Shashidhar            Shetty
ks@gmail.com              515.123.4567         17-JUN-03 AD_PRES          22000

        210 Tejaswini             Aradhya
EMPLOYEE_ID FIRST_NAME            LAST_NAME
----------- --------------------- -------------------------
EMAIL                     PHONE_NUMBER         HIRE_DATE JOB_ID          SALARY
------------------------- -------------------- --------- ---------- ----------
COMMISSION_PCT MANAGER_ID DEPARTMENT_ID
-------------- ---------- -------------
ta@gmail.com              515.123.4567         17-JUN-03 AD_PRES          21000


111 rows selected.
SQL>
```

The above query gives the data details of the "employee" table. But we see that all the data is in different lines which makes it very difficult to analyze. So, we use the following command to see the data in a more orderly fashion.

LINE SIZE:

At the SQL Plus command line, type: set linesize 200 - this will change the line width to 200 characters. You could try a few different line size settings until you find the size that suits you.

SQL > SET LINESIZE 200;
SQL > SELECT * FROM EMPLOYEES;



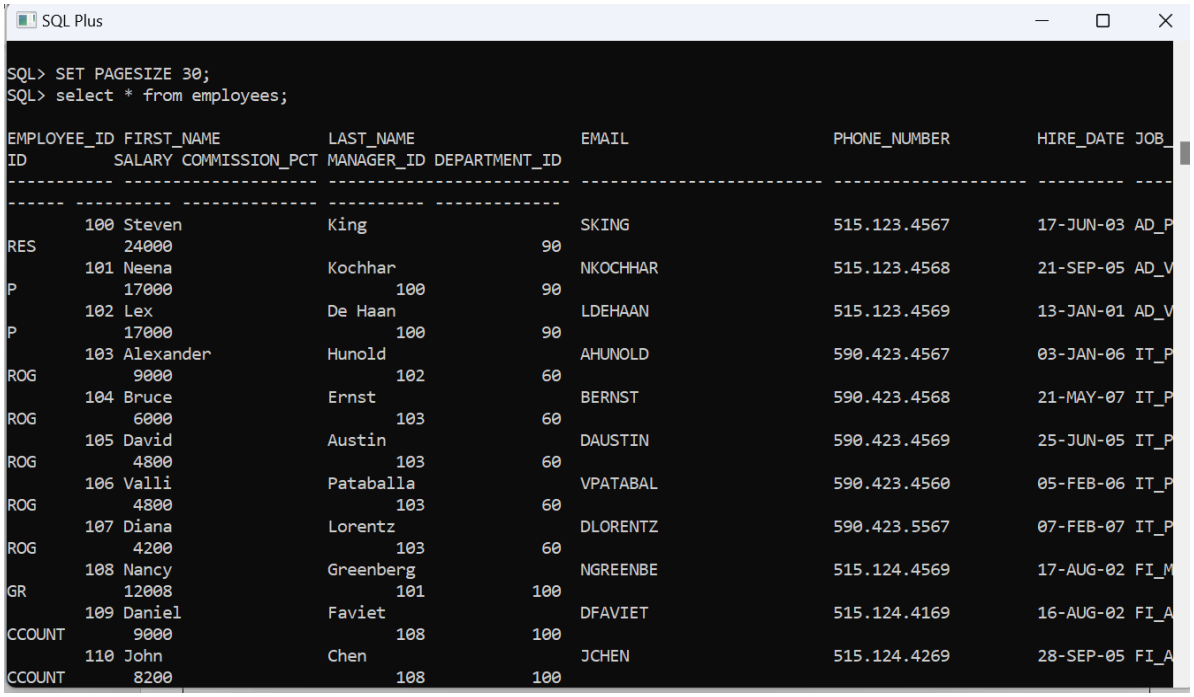
```
EMPLOYEE_ID FIRST_NAME            LAST_NAME                 EMAIL
     PHONE_NUMBER         HIRE_DATE
----------- --------------------- ------------------------- ---------------------
---- -------------------- ----------
JOB_ID          SALARY COMMISSION_PCT MANAGER_ID DEPARTMENT_ID
---------- ---------- -------------- ---------- -------------
        205 Shelley               Higgins                   SHIGGINS
     515.123.8080         07-JUN-02
AC_MGR          12008                        101          110

        206 William               Gietz                     WGIETZ
     515.123.8181         07-JUN-02
AC_ACCOUNT       8300                        205          110

        207 Mohan                 Kumar                     mk@gmail.com
     515.123.4567         17-JUN-03
AD_PRES          26000

        208 Lalitha               Krishna                   pk@gmail.com
     515.123.4567         17-JUN-03
AD_PRES          23000

        209 Shashidhar            Shetty                    ks@gmail.com
     515.123.4567         17-JUN-03
AD_PRES          22000

        210 Tejaswini             Aradhya                   ta@gmail.com
     515.123.4567         17-JUN-03
AD_PRES          21000

111 rows selected.
SQL>
```

The **"set line size"** command helps in increasing the line size; thus, the data is arranged in an orderly fashion.

PAGE SIZE:

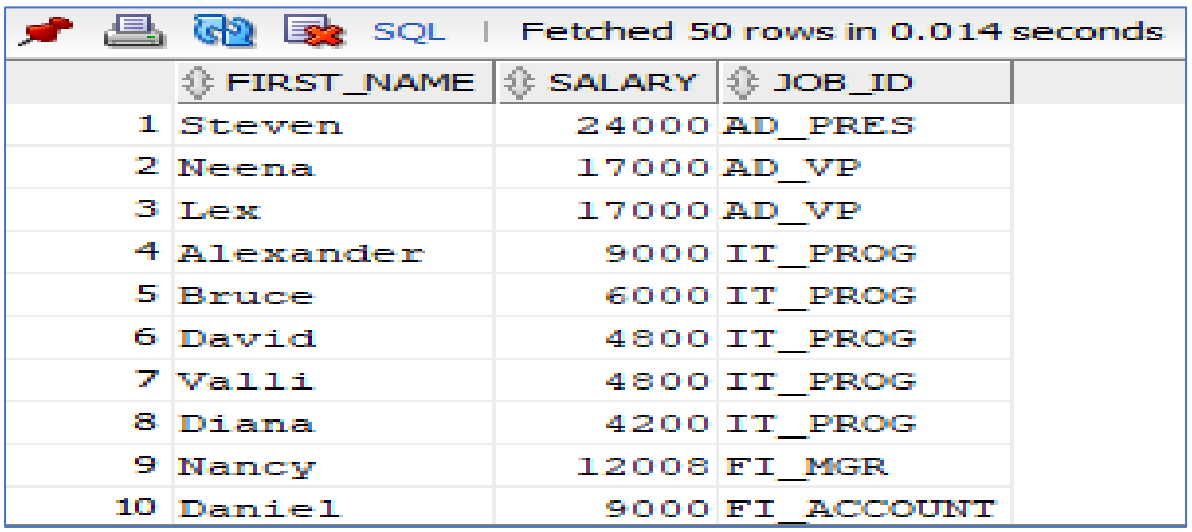
At the SQL Plus command line, type: set pagesize 30 - this will change the page size to 30 rows. set pause on - this will cause the output to pause every 30 lines; press the enter key to continue.

SQL > SET PAGESIZE 30;
SQL > SELECT * FROM EMPLOYEES;



**** To retrieve data for the selected columns of a table?**

SQL > Select first_name, Salary, Job_ID from Employees;



	FIRST_NAME	SALARY	JOB_ID
1	Steven	24000	AD_PRES
2	Neena	17000	AD_VP
3	Lex	17000	AD_VP
4	Alexander	9000	IT_PROG
5	Bruce	6000	IT_PROG
6	David	4800	IT_PROG
7	Valli	4800	IT_PROG
8	Diana	4200	IT_PROG
9	Nancy	12008	FI_MGR
10	Daniel	9000	FI_ACCOUNT

The above query gives the value of only these 3 columns from the table “Employees”.

SQL > Select * from Employees where Salary = 3000;

SQL | All Rows Fetched: 2 in 0.01 seconds

	EMPLOYEE_ID	FIRST_NAME	LAST_NAME	EMAIL	PHONE_NUMBER	HIRE_DATE	JOB_ID	SALARY	COMMISSION_PCT	MANAGER_ID	DEPARTMENT_ID
1	187	Anthony	Cabrio	ACABRIO	650.509.4876	07-02-07	SH_CLERK	3000	(null)	121	50
2	197	Kevin	Feeney	KFEENEY	650.507.9822	23-05-06	SH_CLERK	3000	(null)	124	50

"where" clause is used to restrict the number of records displayed. It gives only the records of the specified condition.

SQL > Select * from Employees where Job_id = 'IT_PROG';

SQL | All Rows Fetched: 5 in 0.008 seconds

	EMPLOYEE_ID	FIRST_NAME	LAST_NAME	EMAIL	PHONE_NUMBER	HIRE_DATE	JOB_ID	SALARY	COMMISSION_PCT	MANAGER_ID	DEPARTMENT_ID
1	103	Alexander	Hunold	AHUNOLD	590.423.4567	03-01-06	IT_PROG	9000	(null)	102	60
2	104	Bruce	Ernst	BERNST	590.423.4568	21-05-07	IT_PROG	6000	(null)	103	60
3	105	David	Austin	DAUSTIN	590.423.4569	25-06-05	IT_PROG	4800	(null)	103	60
4	106	Valli	Pataballa	VPATABAL	590.423.4560	05-02-06	IT_PROG	4800	(null)	103	60
5	107	Diana	Lorentz	DLORENTZ	590.423.5567	07-02-07	IT_PROG	4200	(null)	103	60

Any string data should be enclosed within single quotes (') and the same becomes case sensitive.

ASSIGNMENT:

1) List the employees in dept 20.
2) List the employees earning more than Rs 2500.
3) Display all Salesman.
4) Show the structure of the DEPT table? Also Select all data from the DEPT table?

1) List the employees in dept 20

SQL > Select * from Employees where Department_id = 20;

SQL | All Rows Fetched: 2 in 0.005 seconds

	EMPLOYEE_ID	FIRST_NAME	LAST_NAME	EMAIL	PHONE_NUMBER	HIRE_DATE	JOB_ID	SALARY	COMMISSION_PCT	MANAGER_ID	DEPARTMENT_ID
1	201	Michael	Hartstein	MHARTSTE	515.123.5555	17-02-04	MK_MAN	13000	(null)	100	20
2	202	Pat	Fay	PFAY	603.123.6666	17-08-05	MK_REP	6000	(null)	201	20

2) List the employees earning more than Rs 2500.

SQL > Select * from Employees where Salary > 2500;

SQL | Fetched 50 rows in 0.005 seconds

	EMPLOYEE_ID	FIRST_NAME	LAST_NAME	EMAIL	PHONE_NUMBER	HIRE_DATE	JOB_ID	SALARY	COMMISSION_PCT	MANAGER_ID	DEPARTMENT_ID
1	100	Steven	King	SKING	515.123.4567	17-06-03	AD_PRES	24000	(null)	(null)	90
2	101	Neena	Kochhar	NKOCHHAR	515.123.4568	21-09-05	AD_VP	17000	(null)	100	90
3	102	Lex	De Haan	LDEHAAN	515.123.4569	13-01-01	AD_VP	17000	(null)	100	90
4	103	Alexander	Hunold	AHUNOLD	590.423.4567	03-01-06	IT_PROG	9000	(null)	102	60
5	104	Bruce	Ernst	BERNST	590.423.4568	21-05-07	IT_PROG	6000	(null)	103	60
6	105	David	Austin	DAUSTIN	590.423.4569	25-06-05	IT_PROG	4800	(null)	103	60
7	106	Valli	Pataballa	VPATABAL	590.423.4560	05-02-06	IT_PROG	4800	(null)	103	60
8	107	Diana	Lorentz	DLORENTZ	590.423.5567	07-02-07	IT_PROG	4200	(null)	103	60
9	108	Nancy	Greenberg	NGREENBE	515.124.4569	17-08-02	FI_MGR	12008	(null)	101	100
10	109	Daniel	Faviet	DFAVIET	515.124.4169	16-08-02	FI_ACCOUNT	9000	(null)	108	100

3) Display all Salesman

SQL > Select * from Employees where Job_id = 'SA_MAN';

SQL | All Rows Fetched: 5 in 0.003 seconds

	EMPLOYEE_ID	FIRST_NAME	LAST_NAME	EMAIL	PHONE_NUMBER	HIRE_DATE	JOB_ID	SALARY	COMMISSION_PCT	MANAGER_ID	DEPARTMENT_ID
1	145	John	Russell	JRUSSEL	011.44.1344.429268	01-10-04	SA_MAN	14000	0.4	100	80
2	146	Karen	Partners	KPARTNER	011.44.1344.467268	05-01-05	SA_MAN	13500	0.3	100	80
3	147	Alberto	Errazuriz	AERRAZUR	011.44.1344.429278	10-03-05	SA_MAN	12000	0.3	100	80
4	148	Gerald	Cambrault	GCAMBRAU	011.44.1344.619268	15-10-07	SA_MAN	11000	0.3	100	80
5	149	Eleni	Zlotkey	EZLOTKEY	011.44.1344.429018	29-01-08	SA_MAN	10500	0.2	100	80

4) Show the structure of the DEPT table? Also Select all data from the DEPT table?

SQL > DESC DEPARTMENTS;

Task completed in 0.19 seconds

```
Name                 Null?    Type
-------------------- -------- ------------
DEPARTMENT_ID        NOT NULL NUMBER(4)
DEPARTMENT_NAME      NOT NULL VARCHAR2(30)
MANAGER_ID                    NUMBER(6)
LOCATION_ID                   NUMBER(4)
Name                 Null?    Type
-------------------- -------- ------------
```

SQL > SELECT * FROM DEPARTMENTS;

```
5  SELECT * FROM DEPARTMENTS;
6
```

Script Output | Query Result

SQL | All Rows Fetched: 27 in 0.01 seconds

	DEPARTMENT_ID	DEPARTMENT_NAME	MANAGER_ID	LOCATION_ID
1	10	Administration	200	1700
2	20	Marketing	201	1800
3	30	Purchasing	114	1700
4	40	Human Resources	203	2400
5	50	Shipping	121	1500
6	60	IT	103	1400
7	70	Public Relations	204	2700
8	80	Sales	145	2500
9	90	Executive	100	1700
10	100	Finance	108	1700
11	110	Accounting	205	1700

****