**PRACTICAL - 1**

**Aim:**

Evaluation of Database (File System, DBMS, RDBMS, DDBMS).

**File System:­­­­­­­­­­­**

* A File Management system is a DBMS that allows acceSs to single files or tables at a time. In a File System, data is directly stored in set of files. It contains flat files that have no relation to other files (when only one table is stored in single file, then this file is known as flat file).
* It helps you to organizes the data and allows easy retrieval of files when they are required. It mostly consists of different types of files like mp3, mp4, txt, doc, etc. that are grouped into directories.
* A file system enables you to handle the way of reading and writing data to the storage medium. It is directly installed into the computer with the Operating systems such as Windows and

**Features of a File system:**

Here are important elements of the file system:

* It helps you to store data in a group of files.
* Files data are dependent on each other.
* C/C++ and COBOL languages were used to design the files.
* Shared File System Support
* Fast File System Recovery.

**DBMS:**

* A Database Management System (DBMS) is software designed to store, retrieve, define, and manage data in a database.
* DBMS software primarily functions as an interface between the end user and the database, simultaneously managing the data, the database engine, and the database schema in order to facilitate the organization and manipulation of data.

**Features of DBMS:**

* A user-accessible catalog of data
* Transaction support
* Concurrency control with Recovery services
* Authorization services
* The value of data is the same at all places.
* Offers support for data communication
* Independent utility services
* Allows multiple users to share a file at the same time



* DBMS software primarily functions as an interface between the end user and the database, simultaneously managing the data, the database engine, and the database schema in order to facilitate the organization and manipulation of data.
* Though functions of DBMS vary greatly, general-purpose DBMS features and capabilities should include: a user accessible catalogue describing metadata, DBMS library management system, data abstraction and independence, data security, logging and auditing of activity, support for concurrency and transactions, support for authorization of access, access support from remote locations, DBMS data recovery support in the event of damage, and enforcement of constraints to ensure the data follows certain rules.

**RDBMS:**

* A relational database management system (RDBMS) refers to a collection of programs and capabilities that is designed to enable the user to create, update, and administer a [relational database](https://www.omnisci.com/technical-glossary/relational-database), which is characterized by its structuring of data into logically independent tables, normally including a Structured Query Language (SQL) application programming interface
* All modern database management systems like SQL, MS SQL Server, IBM DB2, ORACLE, My-SQL and Microsoft Access are based on RDBMS.
* It is a DBMS in which the database is organized and accessed according to the relationships between data items. In a relational database, relationships between data items are expressed by means of tables. Interdependencies among these tables are expressed by data values rather than by pointers. This allows a high degree of data independence.

**DDBMS:**

* A distributed database management system (DDBMS) is a set of multiple, logically interrelated databases distributed over a network. They provide a mechanism that makes the distribution of data transparent to users.
* Databases in the collection are logically interrelated with each other. Often they represent a single logical database.
* Data is physically stored across multiple sites. Data in each site can be managed by a DBMS independent of the other sites.
* The processors in the sites are connected via a network. They do not have any multiprocessor configuration.

**DIFFERENCE**

**1.File System vs DBMS**

|  |  |
| --- | --- |
| **File System** | **DBMS** |
| File system is a software that manages and organizes the files in a storage medium within a computer**.** | DBMS is a software for managing the database. |
| It doesn’t provide backup and recovery of data if it is lost. | It provides backup and recovery of data even if it is lost. |
| Redundant data can be present in a it. | In it there is no redundant data. |
| There is less data consistency in file system | There is more data consistency because of the process of normalization. |
| There is no efficient query processing in file system. | Efficient query processing is there in DBMS. |
| There is no data independence. | In DBMS data independence exists. |
| File systems provide less security in comparison to DBMS. | DBMS has more security mechanisms as compared to file system. |

**2. DBMS VS RDBMS**

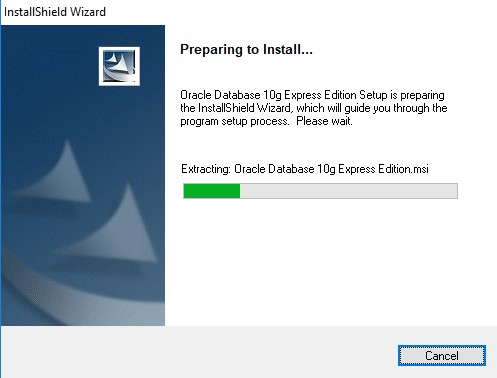
|  |  |
| --- | --- |
| **DBMS** | **RDBMS** |
| DBMS stores data as file. | RDBMS stores data in tabular form. |
| Data elements need to access individually. | Multiple data elements can be accessed at the same time. |
| DBMS does not support distributed database. | RDBMS supports distributed database. |
| Normalization is not present. | Normalization is present. |
| It deals with small quantity of data. | It deals with large amount of data. |
| It supports single user. | It supports multiple users. |
| The data in a DBMS is subject to low security levels with regards to data manipulation. | The data in a DBMS is subject to low security levels with regards to data manipulation. |
| Data fetching is slower for the large amount of data. | Data fetching is fast because of relational approach. |
| Examples: XML, Window Registry, etc | Examples: MySQL, PostgreSQL, SQL Server, Oracle, Microsoft Access etc. |

**PRACTICAL - 2**

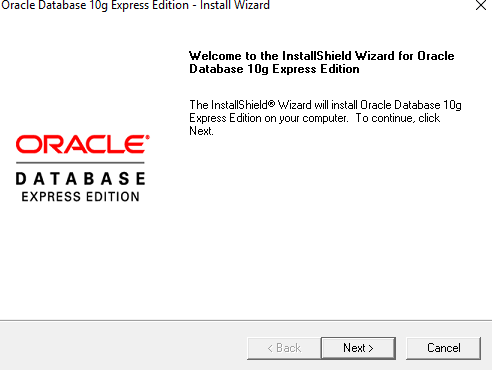
**Aim:**

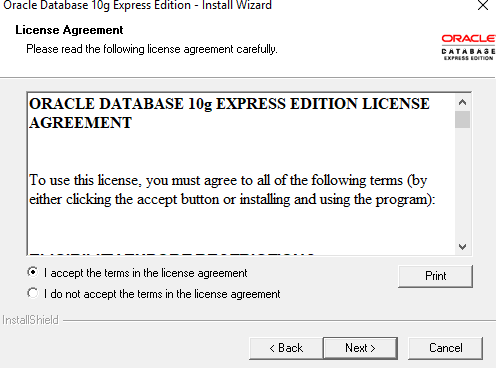
Introduction to Oracle (step by step installation, introduction of sql, pl/sql).

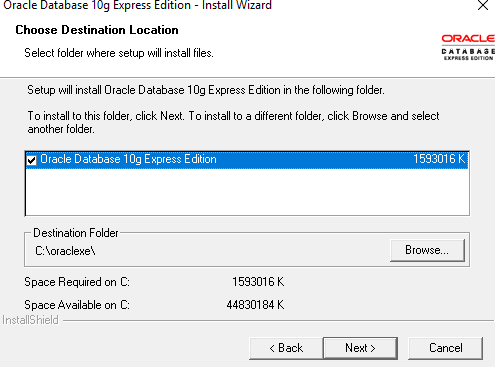
1. **Install it by double clicking .exe which you have downloaded**

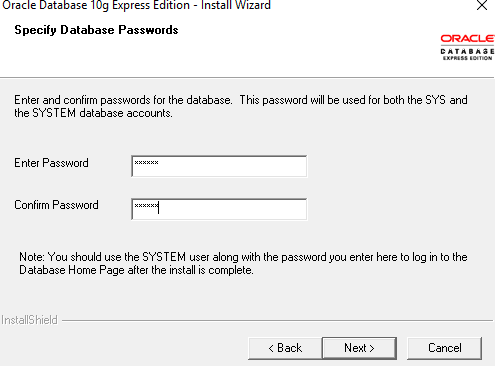
****

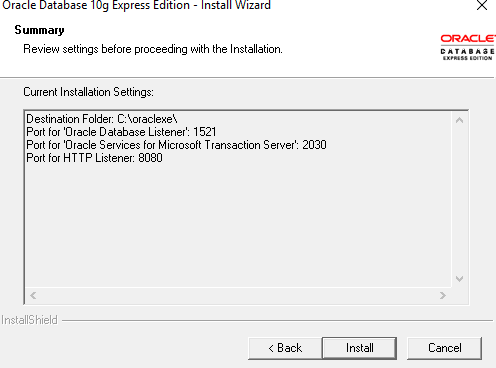
1. **Click on next button**

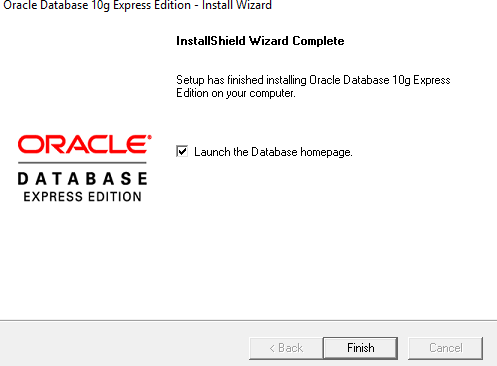
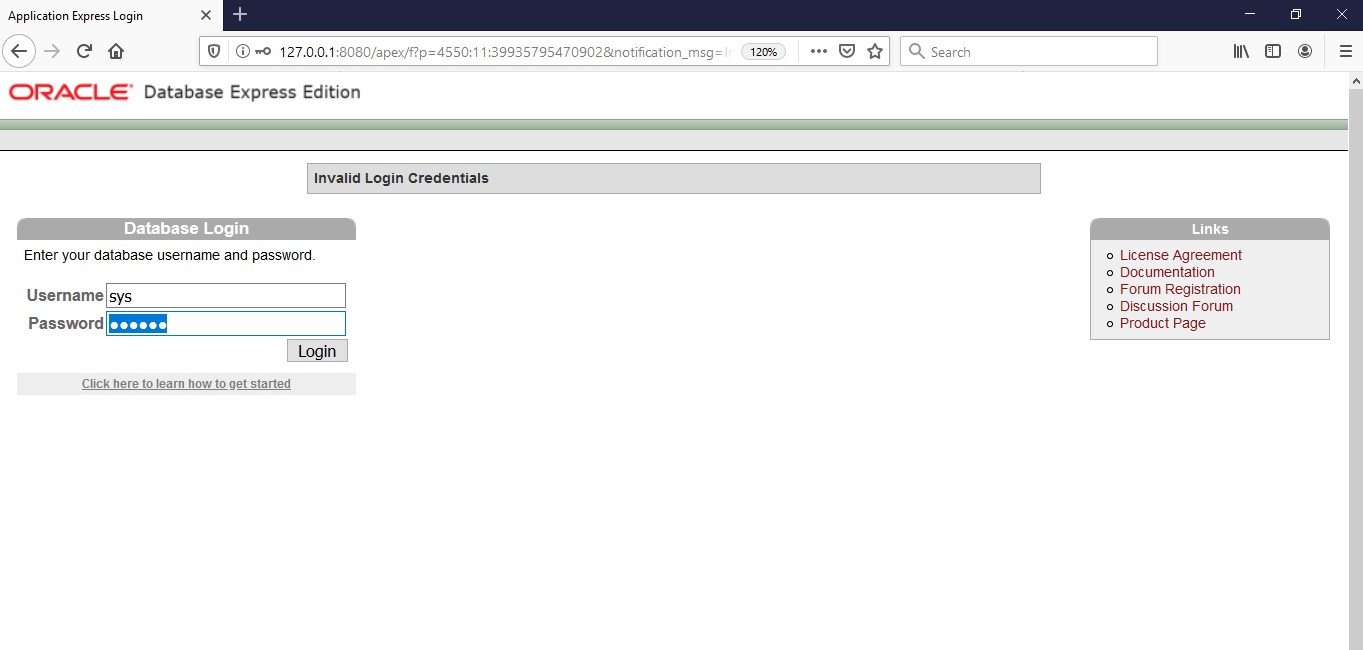
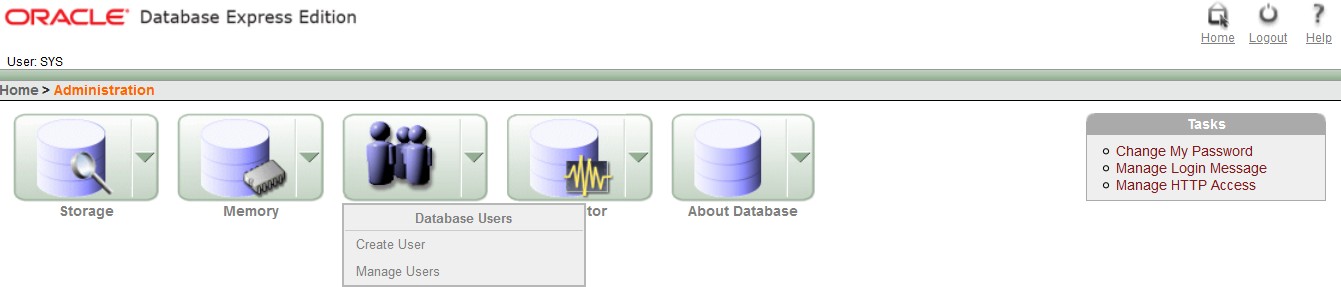
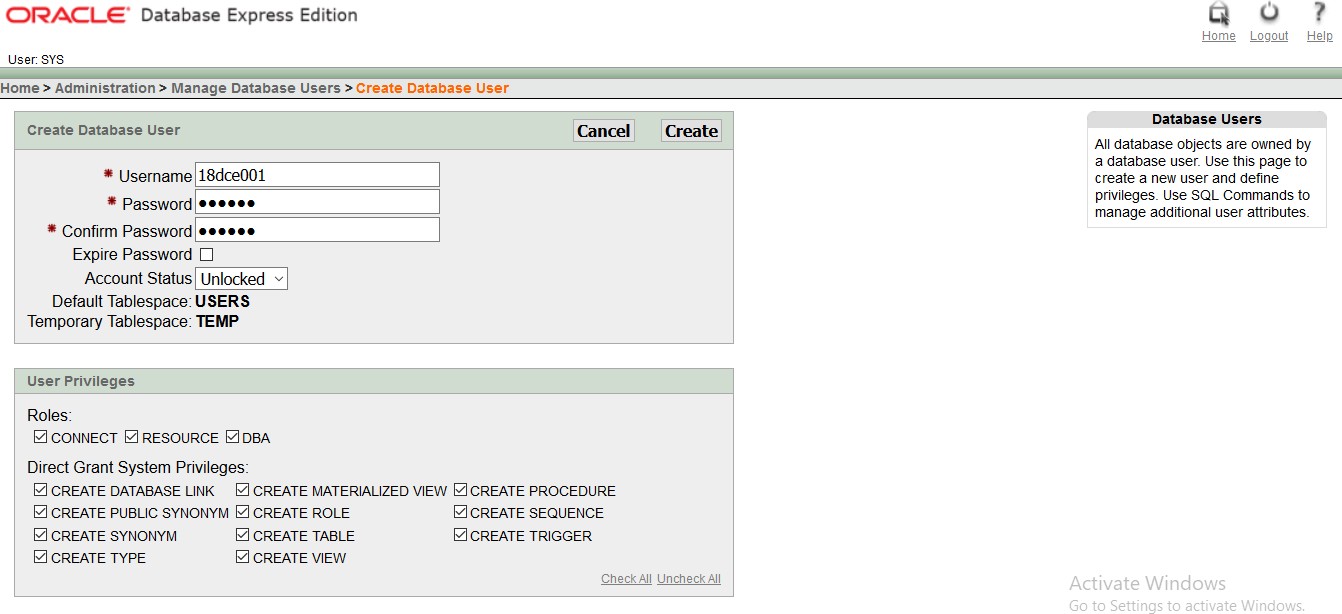
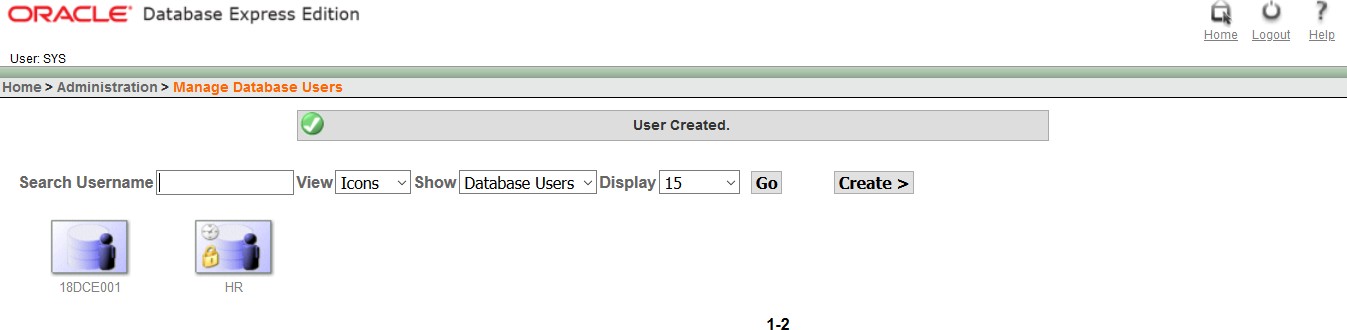
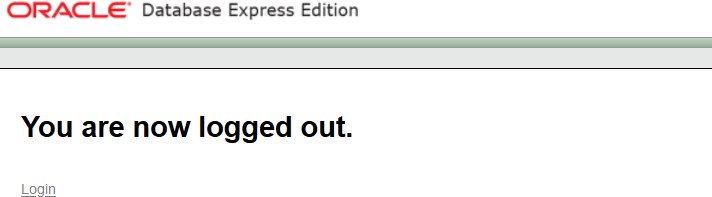
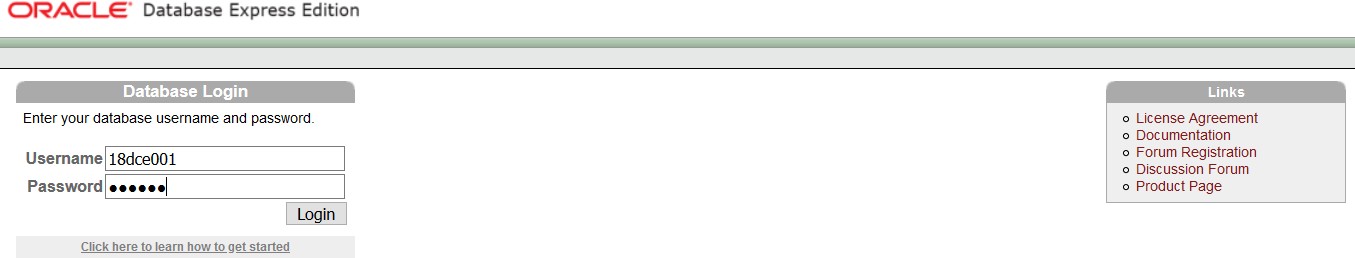
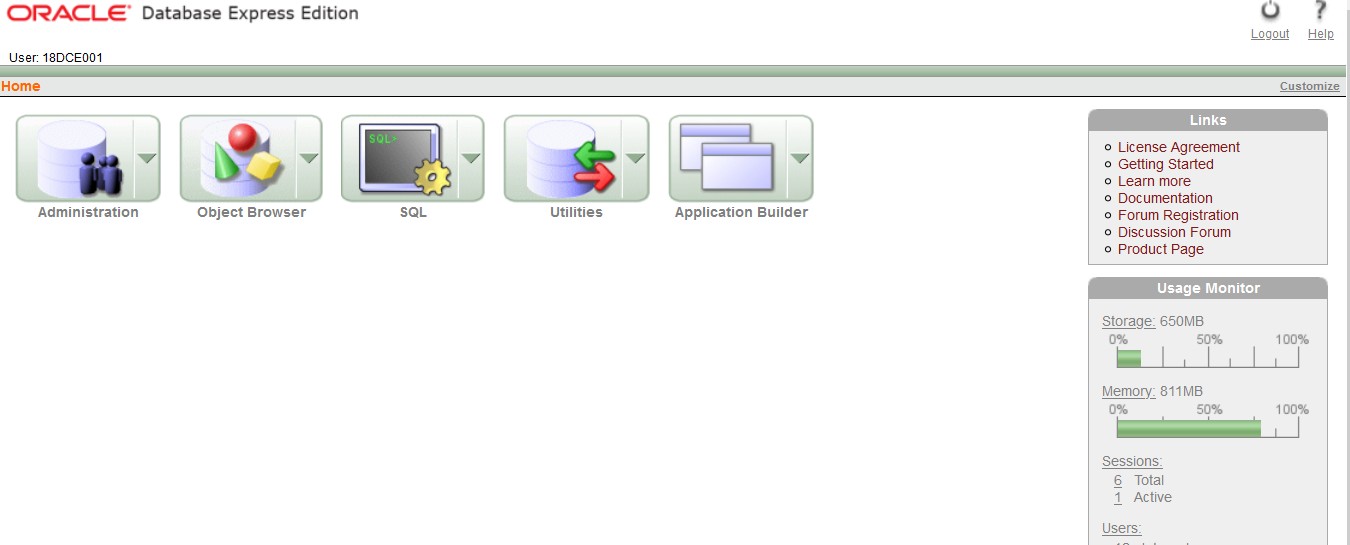
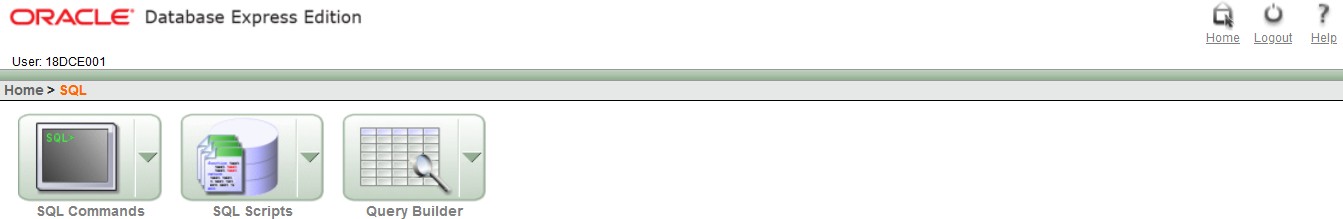
****

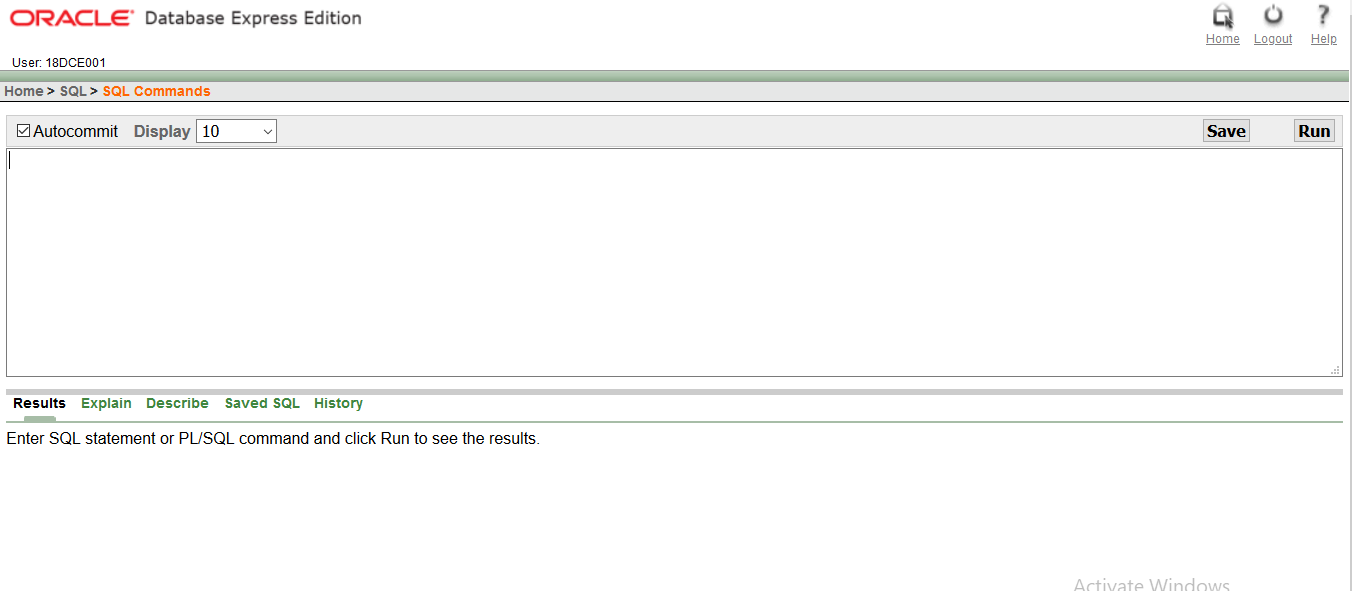
1. **Accept license agreement and click on next button  
     
   **

1. **Click on next button  
     
   **
2. **Enter password and confirm password for SYS and SYSTEM user. Please remember it because once installation will be over you have to enter it. To make it easy to**

**remember give password as : “oracle”  
  
**

1. **Click on install button  
     
   **

1. **Click on finish button.  
     
   **
2. **Enter username as SYS OR SYSTEM and enter your password (Entered in step: 6)  
     
   **
3. **Click on Administration  
     
   **
4. **Now click on “database user drop down button”. From that click on “create user**  
   
5. **Enter your college roll no in username and give password (NEW) and confirm password. Don’t check expire password, make account status unblocked if it is not. Give all privileges to your user. Finally click on “create” button.  
     
   **
6. **This page will be shown to you. Now click on “logout” button.  
     
   **
7. **Click on login  
     
   **
8. **Enter username and password that you just created and click on “login” button  
     
   **
9. **Click on SQL  
     
   **
10. **Home Screen  
      
    **
11. **Congratulation!!! Now you are ready to code SQL and PLSQL.**

****

1. **Thank You**

**Introduction**

**SQL** :

* SQL is Structured Query Language, which is a computer language for storing, manipulating and retrieving data stored in a relational database.
* SQL is the standard language for Relational Database System. All the Relational Database Management Systems (RDMS) like MySQL, MS Access, Oracle, Sybase, Informix, Postgres and SQL Server use SQL as their standard database language.

Also, they are using different dialects, such as −

* MS SQL Server using T-SQL,
* Oracle using PL/SQL,
* MS Access version of SQL is called JET SQL (native format) etc.

**PLSQL :**

* PL/SQL is a block structured language that enables developers to combine the power of SQL with procedural statements. All the statements of a block are passed to oracle engine all at once which increases processing speed and decreases the traffic.

**PRACTICAL – 3**

**AIM:**

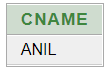
|  |  |
| --- | --- |
| **To study DDL-create and DML-insert commands**.  **(i)** Create tables according to the following definition.   * CREATE TABLE DEPOSIT (ACTNO VARCHAR2(5), CNAME VARCHAR2(18), BNAME VARCHAR2(18), AMOUNT NUMBER (8,2), ADATE DATE); * CREATE TABLE BRANCH (BNAME VARCHAR2(18), CITY VARCHAR2(18));  |  | | --- | | * CREATE TABLE CUSTOMERS (CNAME VARCHAR2(19), CITY VARCHAR2(18)); * CREATE TABLE BORROW (LOANNO VARCHAR2(5), CNAME VARCHAR2(18), BNAME VARCHAR2(18), AMOUNT NUMBER (8,2)); |   **From the above given tables perform the following queries**.  **(1) Describe deposit, branch.**  DESC DEPOSIT;  DESC BRANCH;    **(2) Describe borrow, customers.**  DESC BORROW;    DESC CUSTOMERS;    **(3) List all data from table DEPOSIT.**  SELECT \* FROM DEPOSIT;  **(4) List all data from table BORROW.**  SELECT \* FROM BORROW;  **(5) List all data from table CUSTOMERS.**  SELECT \* FROM CUSTOMERS;  **(6) List all data from table BRANCH.**  SELECT \* FROM BRANCH;    **(7) Give account no and amount of depositors.**  SELECT ACTNO, AMOUNT FROM DEPOSIT;    **(8) Give name of depositors having amount greater than 4000.**  SELECT CNAME FROM DEPOSIT WHERE AMOUNT>4000; |

|  |
| --- |
| **(9) Give name of customers who opened account after date '1-12-96'.** |

**(10) Give name of city where branch karolbagh is located.**SELECT CITY FROM BRANCH WHERE BNAME='KAROLBAGH'; ****

**(11) Give account no and amount of customer having account opened between date**

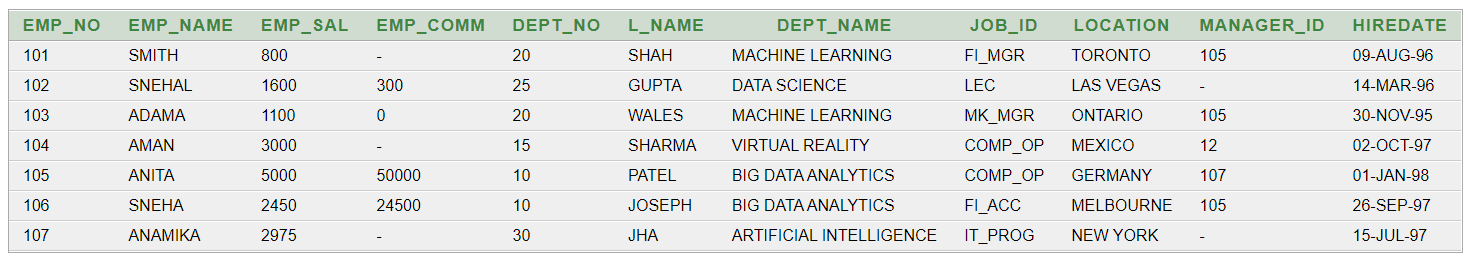
**1-12-96 and 1-6- 96.**

**(12) Give names of depositors having account at VRCE.**   
  
SELECT CNAME FROM DEPOSIT3 WHERE BNAME='VRCE'; ****

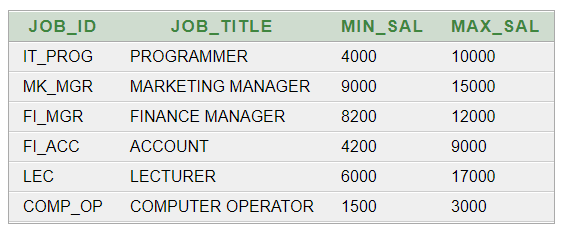
**PRACTICAL - 4**

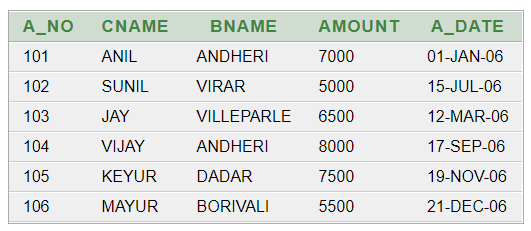
**Aim:**

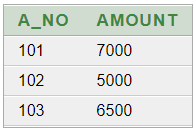
**Create the below given table and insert the data accordingly**.

**(1) Retrieve all data from employee, jobs and deposit.**SELECT \* FROM EMPLOYEE; ****

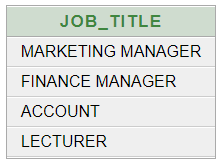
SELECT \* FROM JOB;

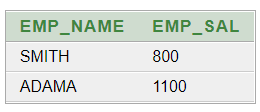


SELECT \* FROM DEPOSIT4;  
  


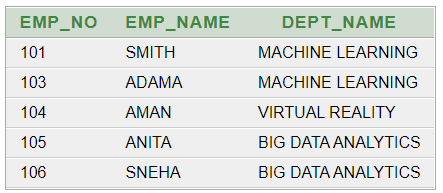
**(2) Give details of account no. and deposited rupees of customers having account opened between dates 01-01-06 and 25-07-06.**   
  
SELECT A\_NO, AMOUNT FROM DEPOSIT4 WHERE A\_DATE BETWEEN '01-JAN-06' AND '25-JUL-06';  
  


**(3) Display all jobs with minimum salary is greater than 4000.**   
  
SELECT JOB\_TITLE FROM JOB WHERE MIN\_SAL > 4000;

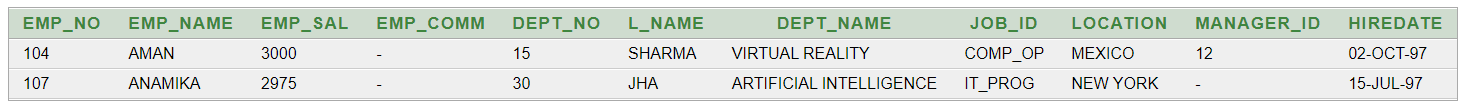


**(4) Display name and salary of employee whose department no is 20. Give alias name to name of employee.**   
  
SELECT EMP\_NAME,EMP\_SAL FROM EMPLOYEE WHERE DEPT\_NO=20;  
  


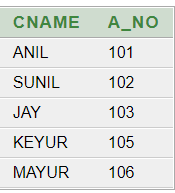
**(5) Display employee no, name and department details of those employee whose department lies in (10,20).**SELECT EMP\_NO, EMP\_NAME, DEPT\_NAME FROM EMPLOYEE WHERE DEPT\_NO BETWEEN '10' AND '20';

****

**(6) Display the non-null values of employees.**SELECT \* FROM EMPLOYEE WHERE EMP\_COMM IS NULL AND EMP\_NAME LIKE '\_\_A%';

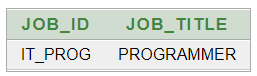
****

**(7) Display name of customer along with its account no (both column should be displayed as one) whose amount is not equal to 8000 Rs.**SELECT CNAME, A\_NO FROM DEPOSIT4 WHERE AMOUNT != '8000';

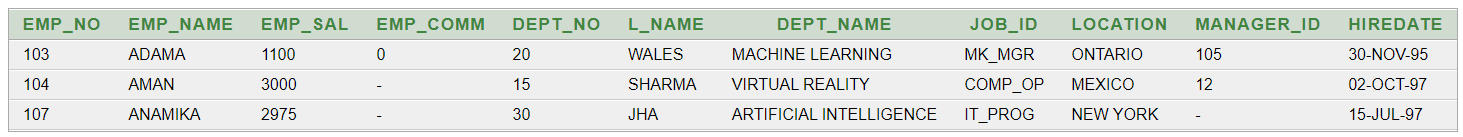
****

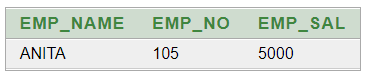
**(8)** **Display the content of job details with minimum salary either 2000 or 4000.**

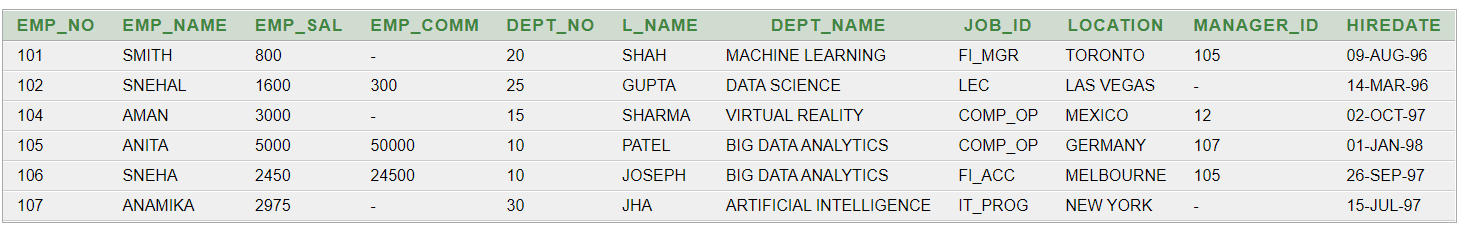
SELECT JOB\_ID,JOB\_TITLE FROM JOB WHERE MIN\_SAL=2000 OR MIN\_SAL=4000;

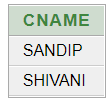


**To study various options of LIKE predicate**

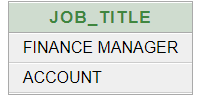
**(1) Display all employee whose name start with ‘A’ and third character is ‘‘a’.**   
  
SELECT \* FROM EMPLOYEE WHERE EMP\_NAME LIKE 'A\_A%';  
  


**(2) Display name, number and salary of those employees whose name is 5 characters long and first three characters are ‘Ani’.**   
  
SELECT EMP\_NAME, EMP\_NO, EMP\_SAL FROM EMPLOYEE WHERE EMP\_NAME LIKE 'ANI\_\_';  
  


**(3) Display all information of employee whose second character of name is either ‘M’ or ‘N’.**   
  
SELECT \* FROM EMPLOYEE WHERE EMP\_NAME LIKE '\_M%' OR EMP\_NAME LIKE '\_N%';  
  


**(4) Find the list of all customer name whose branch is in ‘andheri’ or ‘dadar’ or ‘virar’.**SELECT CNAME FROM DEPOSIT WHERE BNAME IN ('ANDHERI', 'DADAR', 'VIRAR');  
  


**(5) Display the job name whose first three character in job id field is ‘FI\_’.**SELECT JOB\_TITLE FROM JOB WHERE JOB\_ID LIKE 'FI\_%';

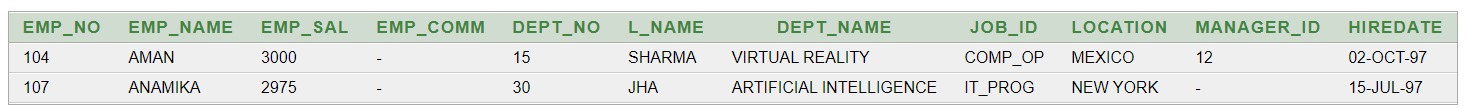
****

**(6) Display the title/name of job who’s last three character are ‘\_MGR’ and their maximum salary is greater than Rs 12000.**SELECT JOB\_TITLE FROM JOB WHERE JOB\_ID LIKE '%\_MGR' AND MAX\_SAL>12000;

****

**(7) Display the non-null values of employees and also employee name second character should be ‘n’ and string should be 5-character long.**

**(8) Display the null values of employee and also employee name’s third character should be ‘a’.**SELECT \* FROM EMPLOYEE WHERE EMP\_COMM IS NULL AND EMP\_NAME LIKE '\_\_A%';

****

**(9) What will be output if you are giving LIKE predicate as ‘%\\_%’ ESCAPE ‘\’**

SELECT \* FROM JOB WHERE JOB\_ID LIKE '%\\_%' ESCAPE '\';

