Practical No. 1

Installing ORACLE/MYSQL/NOSQL

Introduction to Oracle:

Oracle is a multi-model database management system primarily designed for running online transaction processing (OLTP), data warehousing (DW), and mixed (OLTP & DW) database workloads. It was developed by Oracle Corporation and is one of the most widely used databases in enterprise settings. Oracle Database provides a comprehensive and fully integrated stack of cloud applications and platform services, which includes database storage, networking, and application servers.

The database is known for its high performance, robust architecture, scalability, and security, which makes it an ideal choice for handling large amounts of data in industries such as banking, finance, healthcare, and more. It supports SQL (Structured Query Language) and PL/SQL (Procedural Language/SQL) and is highly compatible with enterprise resource planning (ERP) systems.

History of Oracle:

Oracle Corporation was founded in 1977 by Larry Ellison, Bob Miner, and Ed Oates under the original name "Software Development Laboratories" (SDL). The company's first product was a relational database that implemented SQL (Structured Query Language), which at the time was being developed by IBM as part of their System R project. Oracle's SQL-based system became one of the first commercially available databases to use SQL, and it played a crucial role in popularizing this language for database management.

Key milestones in the history of Oracle:

1. 1979: Oracle version 2 was released, the first commercial relational database to use SQL.
2. 1983: The company officially adopted the name Oracle Corporation.
3. 1984: Oracle version 4 introduced read consistency.
4. 1985: Oracle version 5 introduced client-server architecture.
5. 1997: Oracle became the first database to support internet computing.
6. 2004: Oracle launched Oracle 10g, the first database designed for grid computing.
7. 2009: Oracle acquired Sun Microsystems, which included the popular programming language Java.
8. 2013: Oracle 12c was released, designed for cloud computing.
9. 2018: Oracle 18c and 19c were released, emphasizing autonomous features.

- **Oracle 11 g installation link:**

https://www.oracle.com/database/technologies/database-11g-express-edition.html

How to Install the Oracle 11g Database?

To install Oracle 11g, start by running the setup file you downloaded**.**

Step 1.) Extracting Oracle 11g XE Installation Files:

- **Destination Folder:** The files will be extracted to the specified folder, which is "C:\Users\Admin\Downloads\OracleXE112_Win64".
- **Show Extracted Files:** This option, if checked, will display the extracted files in a window once the process is complete.
- **Extract:** Clicking this button will initiate the extraction process.

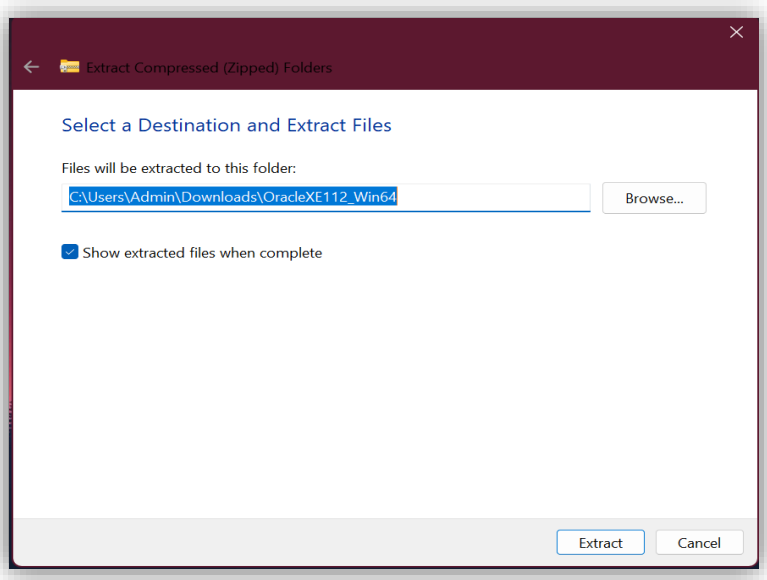

Fig. 1: Shows a dialog box for extracting Oracle 11g XE installation files.

Step 2.) Extracting Files in Progress:

- **Progress Bar:** The green bar indicates the percentage of the extraction that has been completed. In this case, it's 57% complete.
- **Copying Items:** The text "Copying 8 items from OracleXE112_Win64 to OracleXE112_Win64" shows that the extraction is copying files from the source directory to the destination directory.

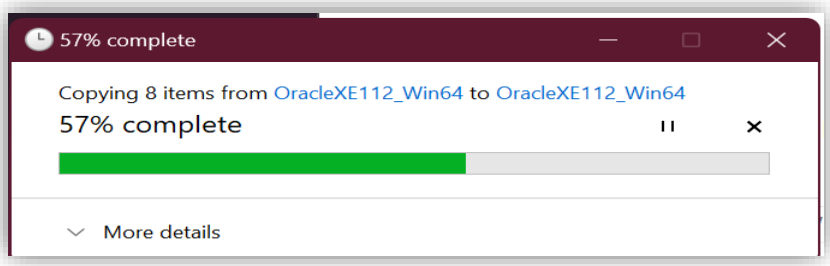

Fig. 2: Shows the progress of the extraction process

Step 3.) Preparing for Installation:

- **Title:** "Oracle Database 11g Express Edition - InstallShield Wizard" indicates that the installation will be guided by the InstallShield Wizard.
- **Message:** "Preparing to Install..." suggests that the setup is getting ready to start the installation process.

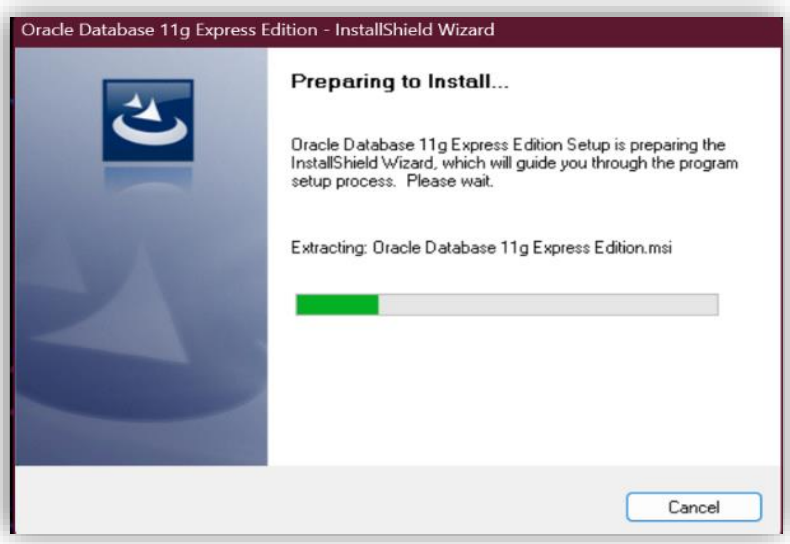

Fig. 3: Shows the initial stage of the Oracle 11g XE installation process

Step 4.) Welcome to the InstallShield Wizard:

- **Title:** "Oracle Database 11g Express Edition - Install Wizard" indicates that we're using the InstallShield Wizard for the installation process.
- **Welcome Message:** The text welcomes us to the wizard and explains its purpose, which is to guide us through the installation of Oracle Database 11g Express Edition.

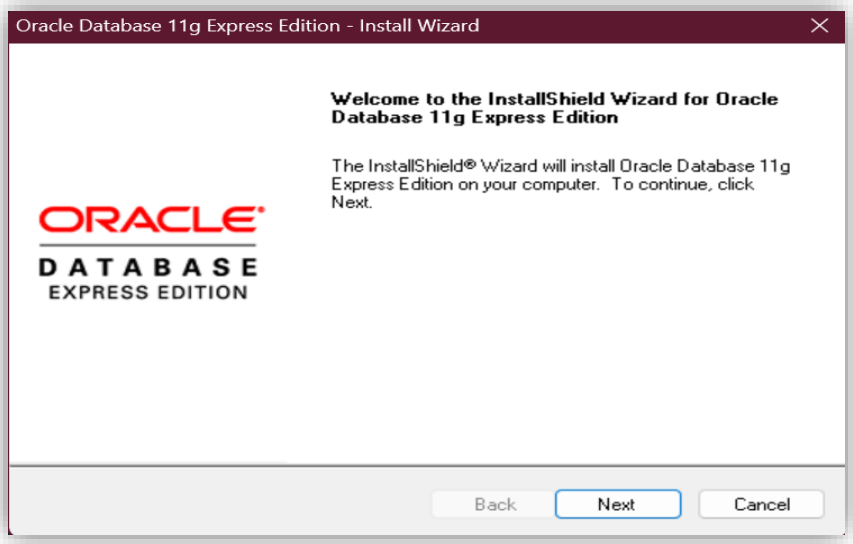

Fig. 4: Shows the welcome screen of the InstallShield Wizard for Oracle Database 11g

Step 5.) License Agreement:

- **Title:** "Oracle Database 11g Express Edition - Install Wizard" indicates that we're still in the installation process.
- **License Agreement:** The text displays the terms and conditions of the license agreement for Oracle Database 11g Express Edition.
- **Acceptance Options:** There are two radio buttons to choose from:
  - "I accept the terms in the license agreement"
  - "I do not accept the terms in the license agreement"

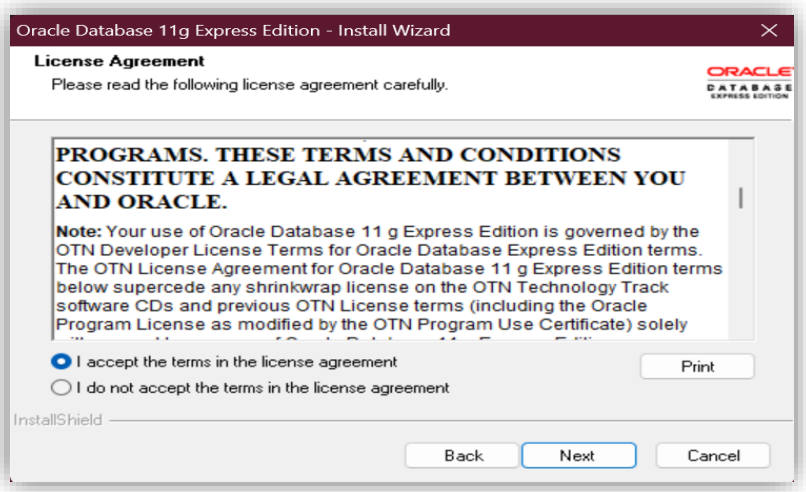

Fig. 5: Shows the license agreement screen of the installation wizard

Step 6.) Choose Destination Location:

- **Title:** "Oracle Database 11g Express Edition - Install Wizard" indicates that we're still in the installation process.
- **Destination Folder:** The default destination folder is shown as "C:\Users\Admin\Desktop\oraclexe".

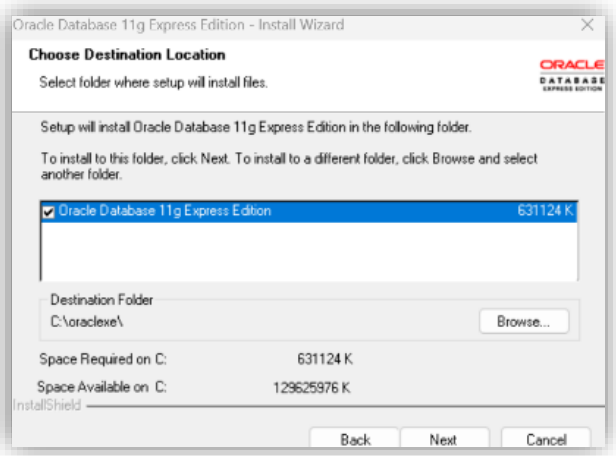

Fig. 6: Shows destination folder for the Oracle Database 11g

Step 7.) Specify Database Passwords:

- **Title:** "Oracle Database 11g Express Edition - Install Wizard" indicates that we're still in the installation process.
- **Password Fields:** There are two password fields:
    - **Enter Password:** This field is for entering the initial password for both the SYS and SYSTEM accounts.
    - **Confirm Password:** This field is for confirming the password entered in the previous field.

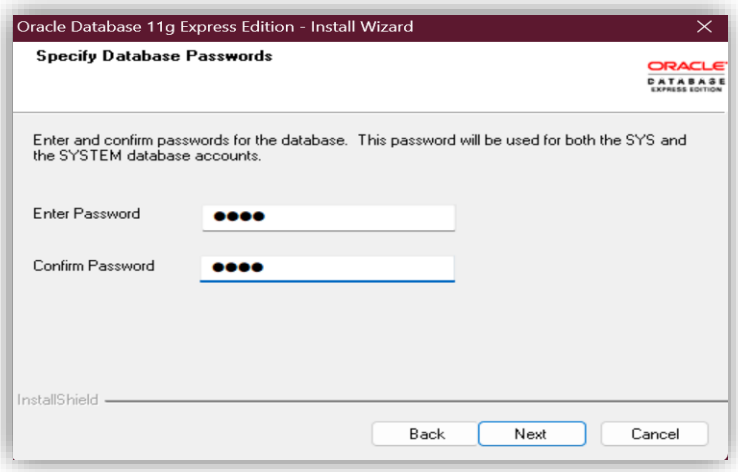

Fig. 7: Shows the screen where we need to set the passwords

Step 8.) Successful Installation:

- **Title:** "Oracle Database XE 11.2" confirms that the installed version is 11.2.
- **Menu Items:** The menu items (Home, Storage, Sessions, Parameters, Application Express) provide access to different components of the database.
- **Links:** The links section provides additional resources for learning and using Oracle Database.
- **News:** The news section might display any relevant updates or announcement

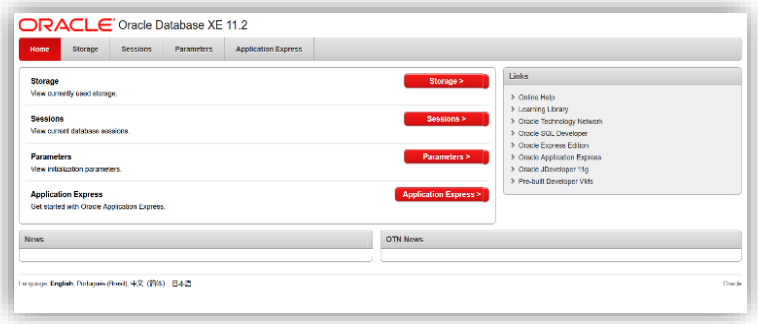

Fig. 8: Shows the Oracle Database 11g Express Edition home page

Step 9.) Oracle Application Express Login Screen:

- **Title:** "Login" indicates that we need to log in to access Oracle Application Express.
- **Username and Password Fields:** We need to enter the username and password for the database user who has been granted the DBA database role. In this case, the username is "sys".

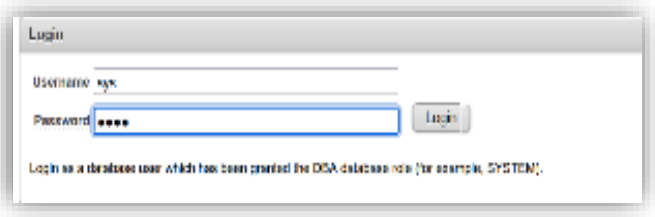

Fig. 9: Shows the login screen for Oracle Application Express

Step 10.) Create Application Express Workspace:

- **Title:** "Oracle Database XE 11.2" indicates that we're still in the Oracle Database environment.
- **Create Application Express Workspace:** This is the main section where we'll create the workspace.
- **Database User Options:** We can choose to create a new database user or use an existing one.
- **Fields:** Provide the database username, Application Express username, password, and confirm password.

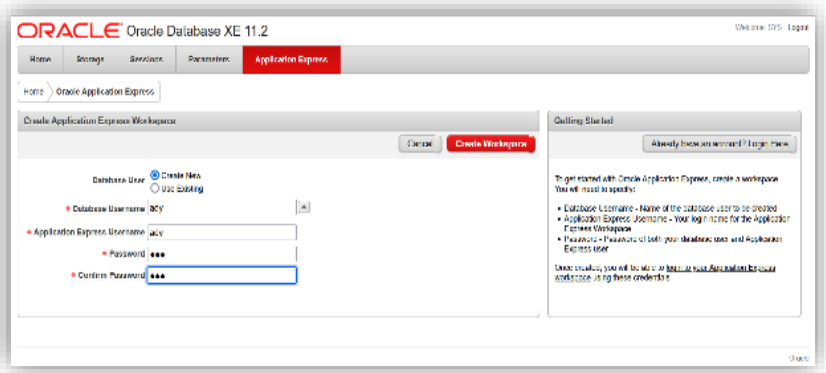

Fig. 10: Shows the screen where we need to create an Application Express workspace

Step 11.) Workspace Creation Success:

- **Title:** "Oracle Database XE 11.2" indicates that we're still in the Oracle Database environment.
- **Success Message:** "Successfully created workspace ADY. To begin, click here to login." confirms that the workspace named "ADY" was created successfully.

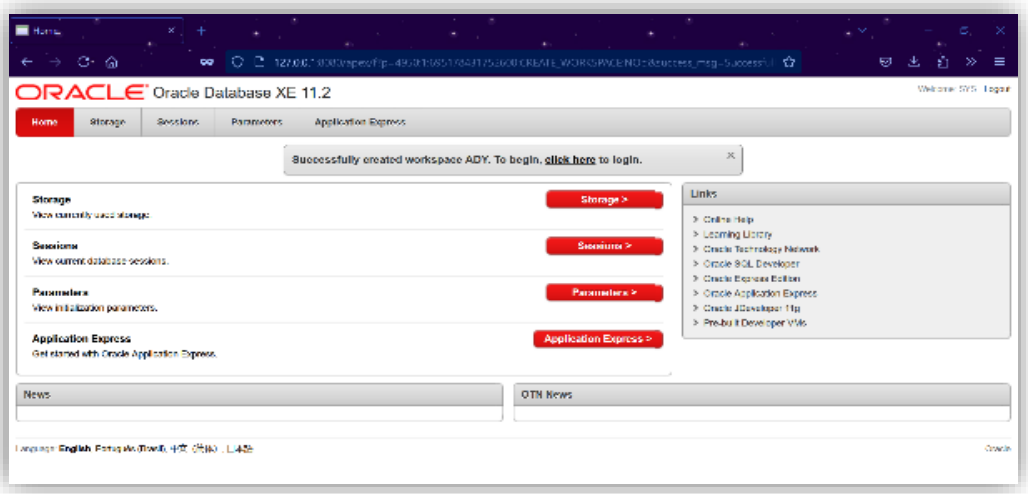

Fig. 11: Shows Application Express workspace has been successfully created

Step 12.) L**ogging into Application Express Workspace:**

- **Title:** "Oracle Application Express" indicates that we're logging into the Application Express environment.
- **Workspace Field:** The "Workspace" field is pre-filled with the name of the workspace we created (ADY).
- **Username and Password Fields:** We need to enter the username (ADY) and password that we specified during the workspace creation process.

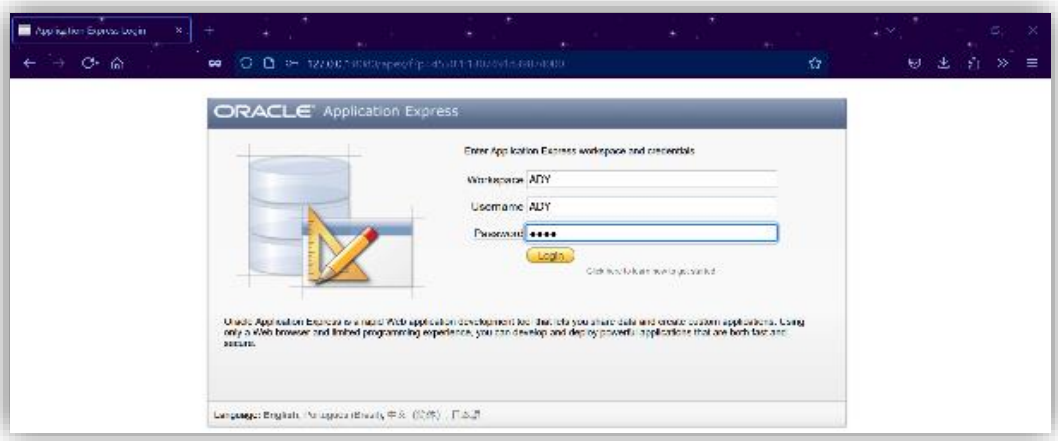

Fig. 12: Shows newly created Application Express workspace

Practical No.2

Creating Entity-Relationship Diagram using case tools with Identifying (entities, attributes, keys and relationships between entities, cardinalities, generalization, specialization etc.)

- **ER Model in Hospital Management System:**

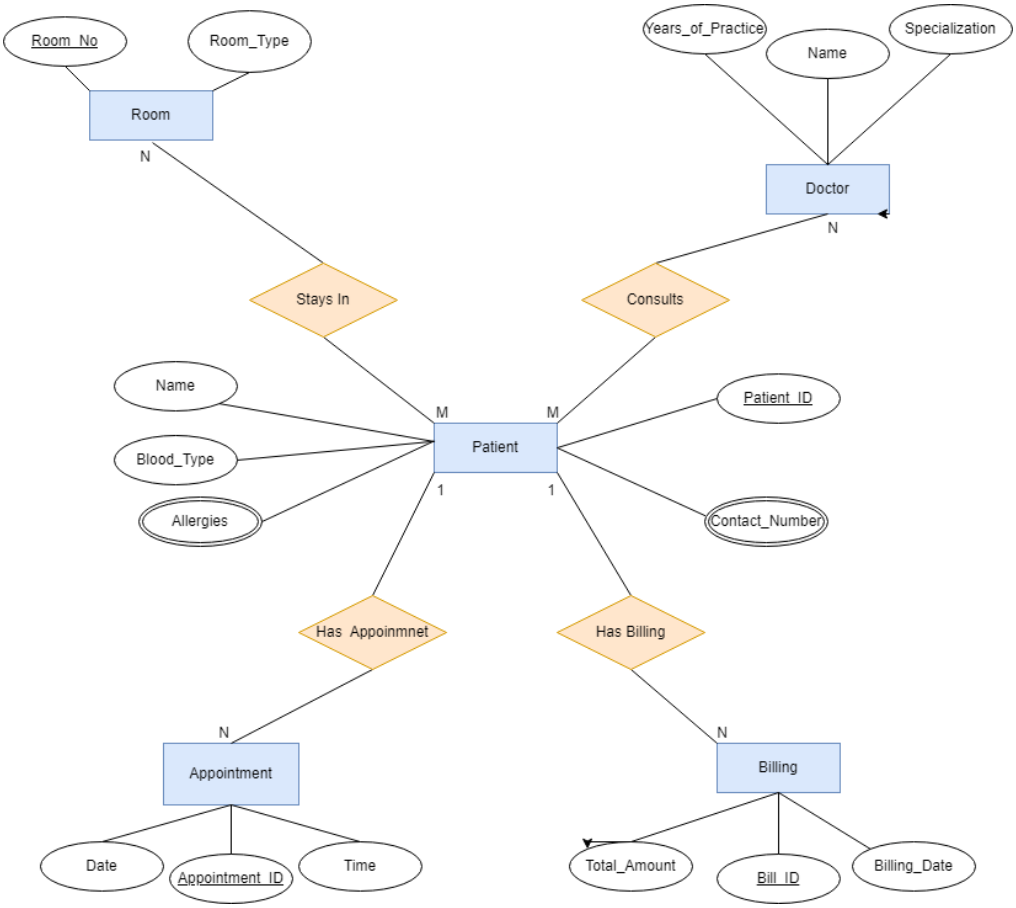

Fig. 14: ER Diagram of Hospital Management System

- **ER Diagram: Visual Representation:**

An ER diagram is a graphical representation of an ER model. It uses symbols to depict entities, attributes, and the relationships between them.

i) **Entity:** Typically represented by a rectangle.
ii) **Attribute:** Usually shown as an oval or circle within the entity rectangle.
iii) **Relationship:** Indicated by a diamond shape connecting two or more entities.

- **Entities and Attributes:**

1. **Patient:** Represents individuals receiving medical care, storing information such as patient ID, name, gender, blood type, contact numbers (which can be multiple), and any allergies.

   Attributes:

   i) Patient_ID (Primary Key)
   ii) Name
   iii) Contact_Number(Multivalued)
   iv) Blood_Type
   v) Allergies (Multivalued)

2. **Doctor:** Contains details of medical professionals, including their ID, name, specialization, contact number, and years of practice.

   Attributes:

   i) Name
   ii) Specialization
   iii) Years_of_Practice

3. **Appointment:** Records the scheduling details of a patient's consultation with a doctor, including the appointment ID, date, and time.

   Attributes:

   i) Appointment_ID (Primary Key)
   ii) Date
   iii) Time

4. **Billing:** Tracks the financial transactions for medical services rendered, storing the bill ID, total amount, billing date, and payment status.

   Attributes:

   i) Bill_ID (Primary Key)
   ii) Total_Amount
   iii) Billing_Date

5. **Room:** Represents hospital rooms available for patients, detailing room number and type (such as single or shared).

   Attributes:

   i) Room_No (Primary Key)
   ii) Room_Type (e.g., Single, Shared)

- **Relationships:**

1. Patient and Doctor:

Relationship: A patient can consult multiple doctors, and a doctor can treat multiple patients.

Cardinality: Many-to-Many relationship.

2. Patient and Appointment:

Relationship: A patient can schedule multiple appointments, and each appointment involves one patient.

Cardinality: One-to-Many relationship.

3. Patient and Billing:

Relationship: A patient can have multiple bills over time for different treatments.

Cardinality: One-to-Many relationship.

4. Patient and Room:

Relationship: A patient may be assigned to multiple rooms during different stages of their treatment, and a room can accommodate multiple patients over time.

Cardinality: Many-to-Many relationship.

Practical No. 3

I. Implement DDL commands –Create, Alter, Drop etc.

Data Definition Language

Data Definition Language (DDL) statements are used to alter/modify a database or table structure and schema. These statements handle the design and storage of database objects. DDL Command by default are auto commit.

1. CREATE - to create objects in the database

Create a employees table with basic details like (employees_id, first_name last,name, age etc)

Create the 'Employees' table

```
CREATE TABLE Employees (
    employee_id INT PRIMARY KEY,
    first_name VARCHAR(50),
    last_name VARCHAR(50),
    department VARCHAR(100),
    salary DECIMAL(10, 2)
);
```

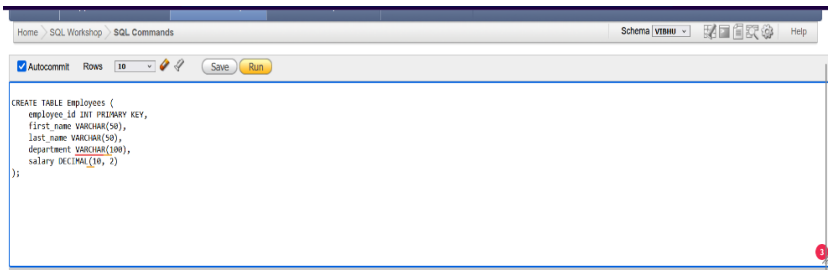

Fig.15: Create the 'Employees' table

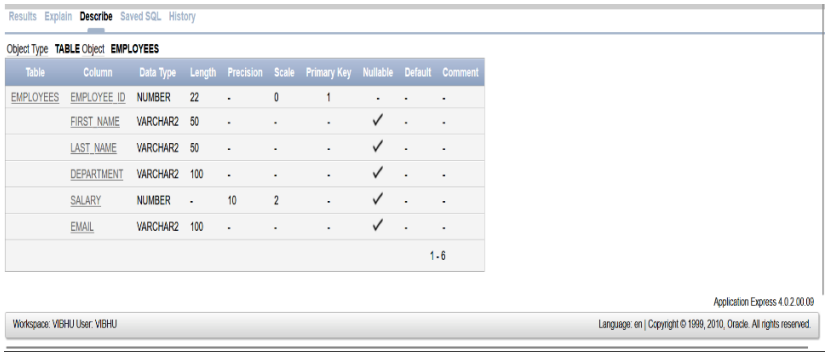

Fig.16: show table

2. ALTER - alters the structure of the database

```
ALTER TABLE Employees
ADD email VARCHAR(100);
```

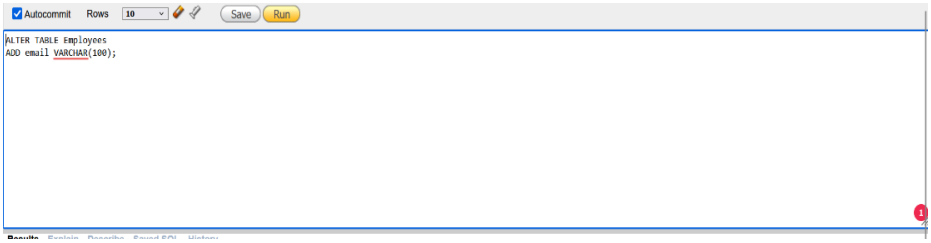

Fig.17: table alter Employees

Used the **DESCRIBE command to show the structure of our table**, such as column names, constraints on column names, etc.

3. TRUNCATE - remove all records from a table, including all spaces allocated for therecords are removed

```
INSERT INTO Employees(employee_id,first_name,last_name,
Department,Salary,email)
VALUES (1, 'Vibhu', 'Gupta', 'CEO', 9909000),
(2, 'John', 'Doe', 'HR', 99090);

TRUNCATE TABLE Employees;
```

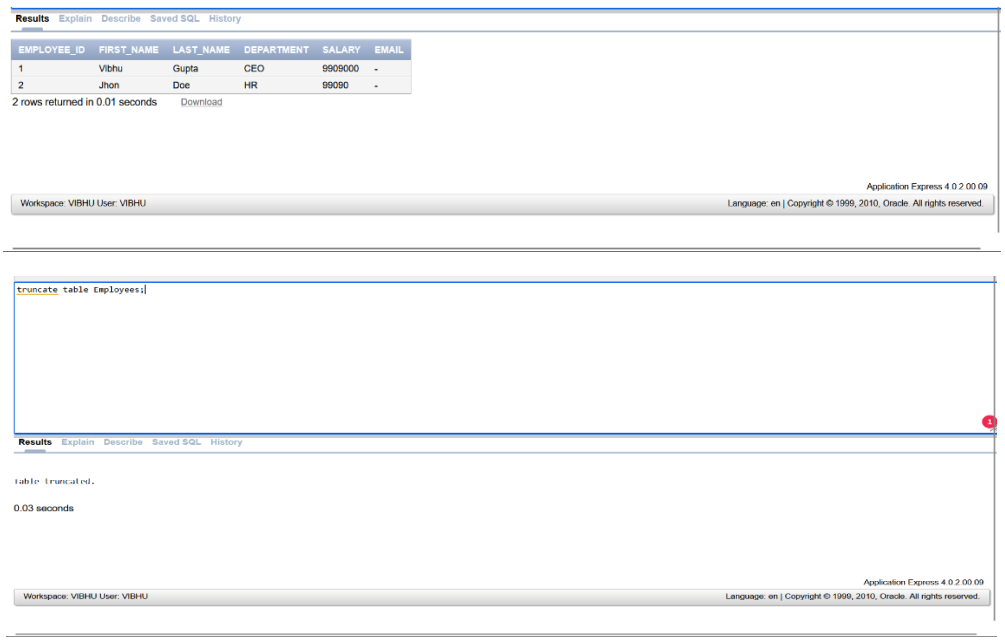

Fig.18: Truncate Employees table

4. DROP - delete the existing table

```
DROP TABLE Employees;
```

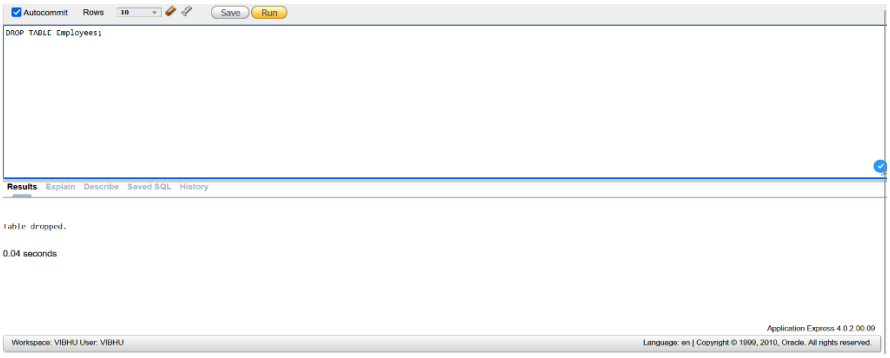

Fig.19: Drop table Employees

II. Implement DML commands- Insert, Select, Update, Delete

Data Manipulation Language

Data Manipulation Language (DML) statements affect records in a table. These are basic operations we perform on data such as selecting a few records from a table, inserting new records, deleting unnecessary records, and updating/modifying existing records.

1. INSERT - insert data into a table

Create a table Courses with basic details like (course_id, course_name, instructor_name, credits)

```
INSERT INTO Employees (employee_id, first_name,
last_name, department, salary, email)
VALUES
 (1, 'Diana', 'Miller', 'IT', 65000.00,
'diana.miller@example.com');
    (2, 'Vibhu', 'Gupta', 'CSE', 905000.00,
'vibhugupta@gmail.com');
SELECT * FROM Employees;
```

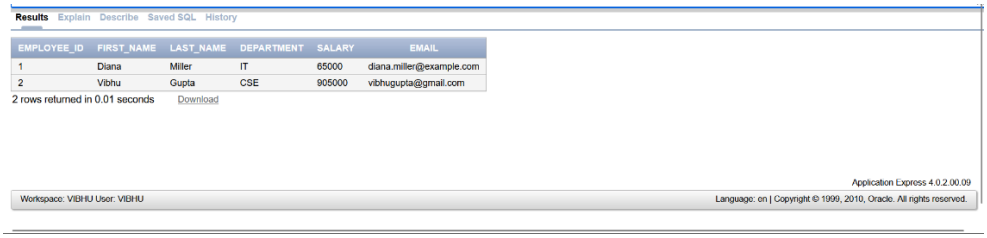

Fig.19: Create a table Courses with basic

2. UPDATE - updates existing data within a table

```
UPDATE Employees
SET salary = 60000.00
WHERE employees_id = 2;
SELECT * FROM Courses;
```

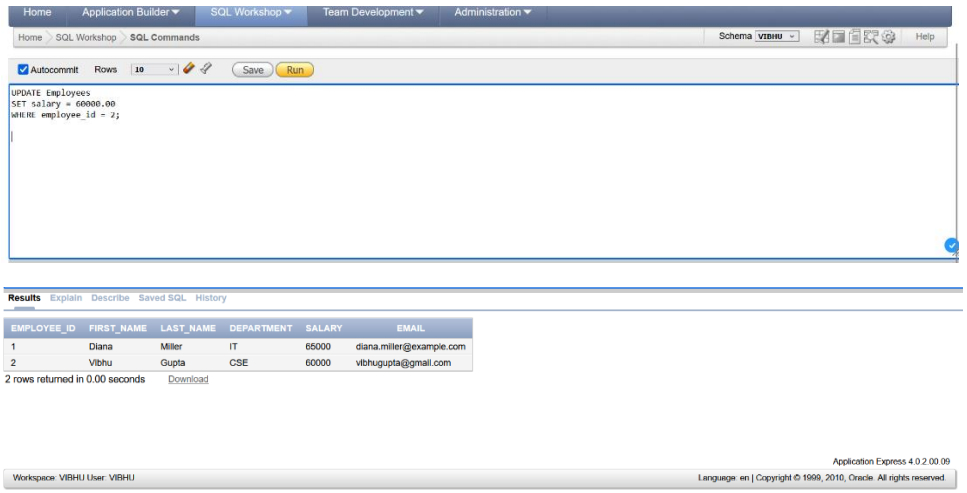

Fig.20: updating existing data within a table

3. DELETE - deletes all records from a table, the space for the records remains

```
DELETE FROM Employees
WHERE employee_id = 1;
SELECT * FROM Employees;
```

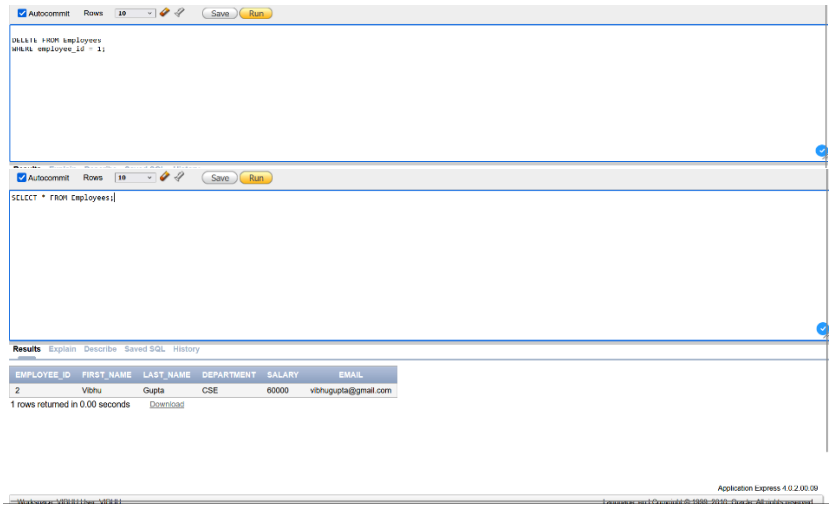

Fig.21: deletes all records from a table

4. SELECT - used to **fetch data from one or more tables**

```
INSERT INTO Employees (employee_id, first_name, last_name, department,
salary, email)

VALUES
(1,'Alice','Johnson','Marketing',50000.00,'alice.johnson@example.com');

(3, 'Charlie','Williams','HR',45000.00,'charlie.williams@example.com');

SELECT first_name, last_name, salary

FROM Employees

ORDER BY salary DESC;
```