وزارة الاتصـــــالات
وتكنولوجيا المعلومات

Databases and SQL for Data Science with Python

Course Overview

If there is a shortcut to becoming a Data Scientist, then learning to think and work like a In this course you will learn SQL inside out- from the very basics of Select statements to advanced concepts like JOINs.

You will:

- write foundational SQL statements like: SELECT, INSERT, UPDATE, and DELETE
- filter result sets, use WHERE, COUNT, DISTINCT, and LIMIT clauses.
- differentiate between DML & DDL
- CREATE, ALTER, DROP and load tables.

Benefits of Enrolling in a Course:

- Analyse data within a database using SQL and Python.
- Create a relational database and work with multiple tables using DDL commands.
- Construct basic to intermediate level SQL queries using DML commands.
- Compose more powerful queries with advanced SQL techniques like views, transactions, stored procedures, and joins.

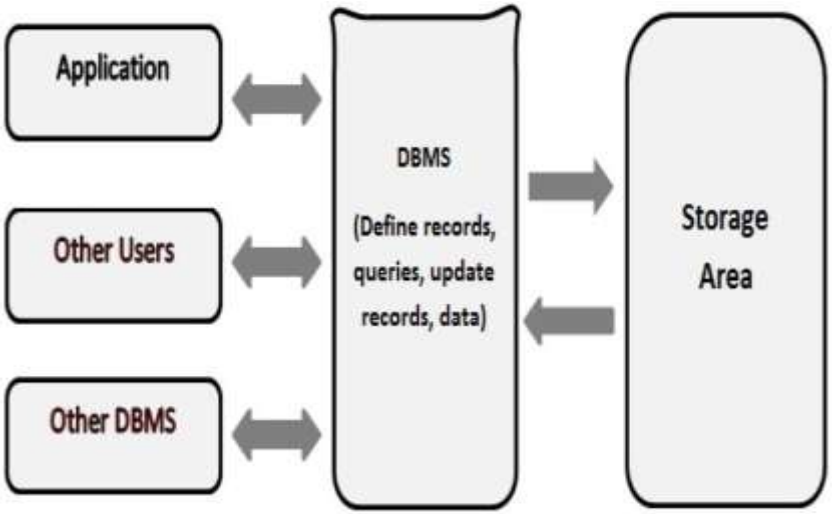
Sessions Content

- Getting started with SQL.
- Introduction to relational database and Tables
- relational database and Tables
- Intermediate SQL
- Intermediate SQL Part 2
- Accessing database using Python
- Accessing database using Python Part 2
- Advanced SQL for Data Engineer
- Advanced SQL for Data Engineer

What are Databases?

- A database is an organized collection of data, generally stored and accessed electronically from a computer system. It supports the storage and manipulation of data.

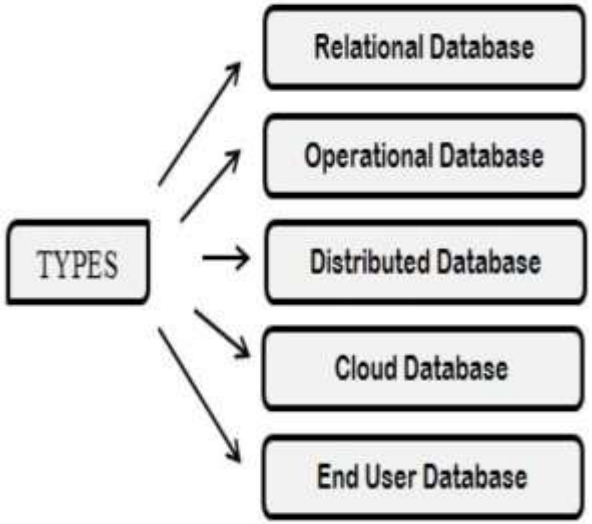
- In other words, databases are used by an organization as a method of storing, managing and retrieving information.



Types of Databases

Depending upon the usage requirements, there are following types of databases available in the market:

- Centralized database
- Distributed database
- Personal database
- End-user database
- Commercial database
- NoSQL database
- Operational database
- Relational database
- Cloud database
- Object-oriented database
- Graph database Here is a detailed article on, Types of Database Management Systems.



Advantages of using Databases

There are many advantages of databases

- Reduced data redundancy
- Reduced updating errors and increased consistency
- Greater data integrity and independence from application programs
- Improved data access to users through the use of host and query languages
- Improved data security
- Reduced data entry, storage, and retrieval costs

Disadvantages of using Databases

There are many disadvantages of databases

- Although databases allow businesses to store and access data efficiently, they also have certain disadvantages
- Complexity
- Cost
- Security
- Compatibility

Some examples of Databases

Some of the most popular databases are

- Oracle Database
- Sybase
- MySQL
- IBM db2

What is SQL?

- SQL (Structured Query Language): Is used to perform operations on the records stored in the database, such as updating records, inserting records, deleting records, creating and modifying database tables, views, etc.

- SQL is not a database system, but it is a query language.

Database Transaction

- A transaction is an executing program that forms a logical unit of database actions.
- It includes one or more database access operations such as insert, delete and update.
- The database operations that form a transaction can either be embedded within an application program or they can be specified interactively via a high-level query language such as SQL.

Database Transaction Properties

- Transactions should possess several properties, often called the ACID properties:

1. Atomicity
2. Consistency
3. Isolation
4. Durability

Database Schema

A schema is a group of related objects in a database. There is one owner of a schema who has access to manipulate the structure of any object in the schema. A schema does not represent a person, although the schema is associated with a user that resides in the database.

Data types

A data type determines the type of data that can be stored in a database column. The most commonly used data types are:

1. Alphanumeric: data types used to store characters, numbers, special characters, or nearly any combination.
2. Numeric
3. Date and Time

Database Constraints

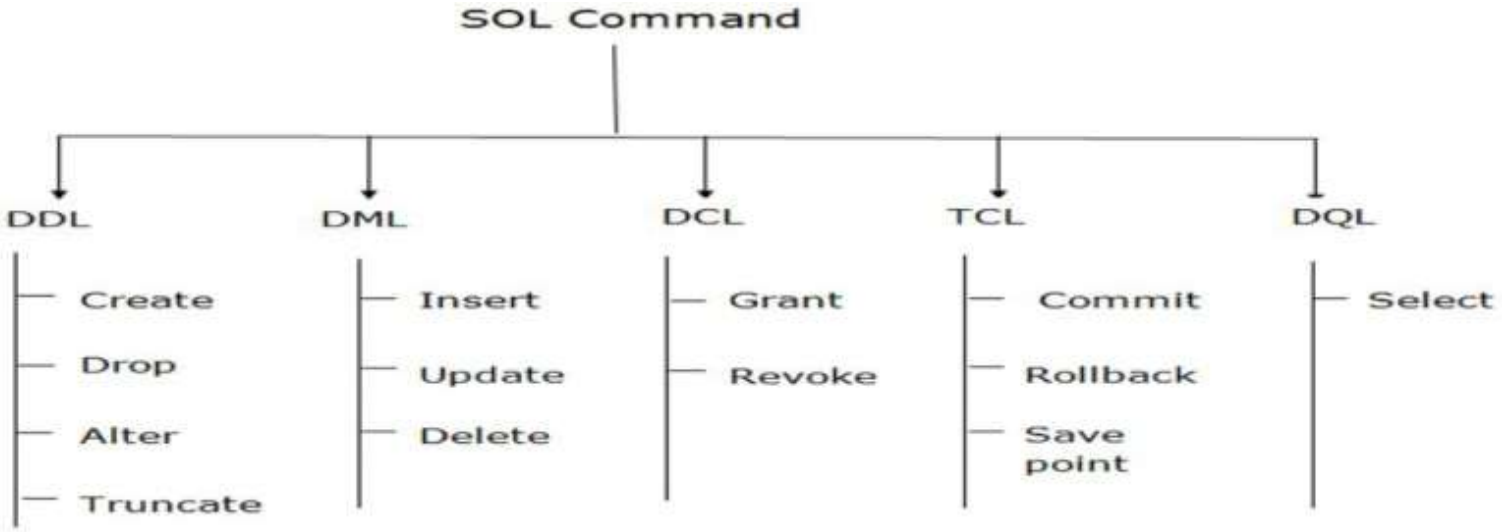
- Primary Key (Not Null + Unique)
- Not Null
- Unique Key
- Referential Integrity (FK)
- Check

SQL Commands

- SQL commands are instructions. It is used to communicate with the database. It is also used to perform specific tasks, functions, and queries of data.

- SQL can perform various tasks like create a table, add data to tables, drop the table, modify the table, set permission for users.

Types of SQL Commands



Data Definition Language (DDL)

- DDL changes the structure of the table like creating a table, deleting a table, altering a table, etc.

- All the command of DDL are auto-committed that means it permanently save all the changes in the database.

Here are some commands that come under DDL:

- CREATE
- ALTER
- DROP
- TRUNCATE

Data Definition Language (DDL)

- CREATE It is used to create a new table in the database

```
CREATE TABLE TABLE_NAME (COLUMN_NAME DATATYPES[,....]);
```

Example

```
CREATE TABLE EMPLOYEE(Name VARCHAR2(20), Email VARCHAR2(100), DOB DATE);
```

Data Definition Language (DDL)

- DROP: It is used to delete both the structure and record stored in the table.

```
DROP TABLE table_name;
```

Example

.

```
DROP TABLE EMPLOYEE;
```

Data Definition Language (DDL)

- ALTER: It is used to alter the structure of the database. This change could be either to modify the characteristics of an existing attribute or probably to add a new attribute.

```
ALTER TABLE table_name ADD column_name COLUMN-definition;
```

Example

.

```
ALTER TABLE STU_DETAILS ADD(ADDRESS VARCHAR2(20));
ALTER TABLE STU_DETAILS MODIFY (NAME VARCHAR2(20));
```

Data Definition Language (DDL)

- TRUNCATE: It is used to delete all the rows from the table and free the space containing the table.

```
TRUNCATE TABLE table_name;
```

Example

```
TRUNCATE TABLE EMPLOYEE;
```

Data Definition Language (DDL)

- TRUNCATE: It is used to delete all the rows from the table and free the space containing the table.

```
TRUNCATE TABLE table_name;
```

Example

```
TRUNCATE TABLE EMPLOYEE;
```

Data Manipulation Language (DML)

- DML commands are used to modify the database. It is responsible for all form of changes in the database.
- The command of DML is not auto-committed that means it can't permanently save all the changes in the database. They can be rollback.

•Here are some commands that come under DML:

- INSERT
- UPDATE
- DELETE

Data Manipulation Language (DML)

- INSERT: The INSERT statement is a SQL query. It is used to insert data into the row of a table

```
INSERT INTO TABLE_NAME
VALUES (value1, value2, value3, .... valueN);
```

Example:

.

```
INSERT INTO javatpoint (Author, Subject) VALUES ("Sonoo", "DBMS");
```

Data Manipulation Language (DML)

- UPDATE: This command is used to update or modify the value of a column in the table.

```
UPDATE table_name SET [column_name1= value1,...column_nameN = valueN] [WHERE CONDITION]
```

Example:

.

```
UPDATE students
SET User_Name = 'Sonoo'
WHERE Student_Id = '3'
```

Data Manipulation Language (DML)

- DELETE: It is used to remove one or more row from a table.

```
DELETE FROM table_name [WHERE condition];
```

Example:

.

```
DELETE FROM javatpoint
WHERE Author="Sonoo";
```

Data Control Language (DCL)

- DCL commands are used to grant and take back authority from any database user.

Here are some commands that come under DCL:

- Grant
- Revoke

Data Control Language (DCL)

- Grant: It is used to give user access privileges to a database.

Example:

```
GRANT SELECT, UPDATE ON MY_TABLE TO SOME_USER, ANOTHER_USER;
```

Data Control Language (DCL)

- Revoke: It is used to take back permissions from the user.

Example:

```
REVOKE SELECT, UPDATE ON MY_TABLE FROM USER1, USER2;
```

Transaction Control Language(TCL)

- TCL commands can only use with DML commands like INSERT, DELETE and UPDATE only.
- These operations are automatically committed in the database that's why they cannot be used while creating tables or dropping them.

Here are some commands that come under TCL:

- COMMIT
- ROLLBACK
- SAVEPOINT

Transaction Control Language(TCL)

- Commit: Commit command is used to save all the transactions to the database.

```
COMMIT;
```

Example

```
DELETE FROM CUSTOMERS
WHERE AGE = 25;
COMMIT;
```

Transaction Control Language(TCL)

- Rollback: Rollback command is used to undo transactions that have not already been saved to the database.

```
ROLLBACK;
```

Example

```
DELETE FROM CUSTOMERS
WHERE AGE = 25;
ROLLBACK;
```

Transaction Control Language(TCL)

- SAVEPOINT: It is used to roll the transaction back to a certain point without rolling back the entire transaction.

Example

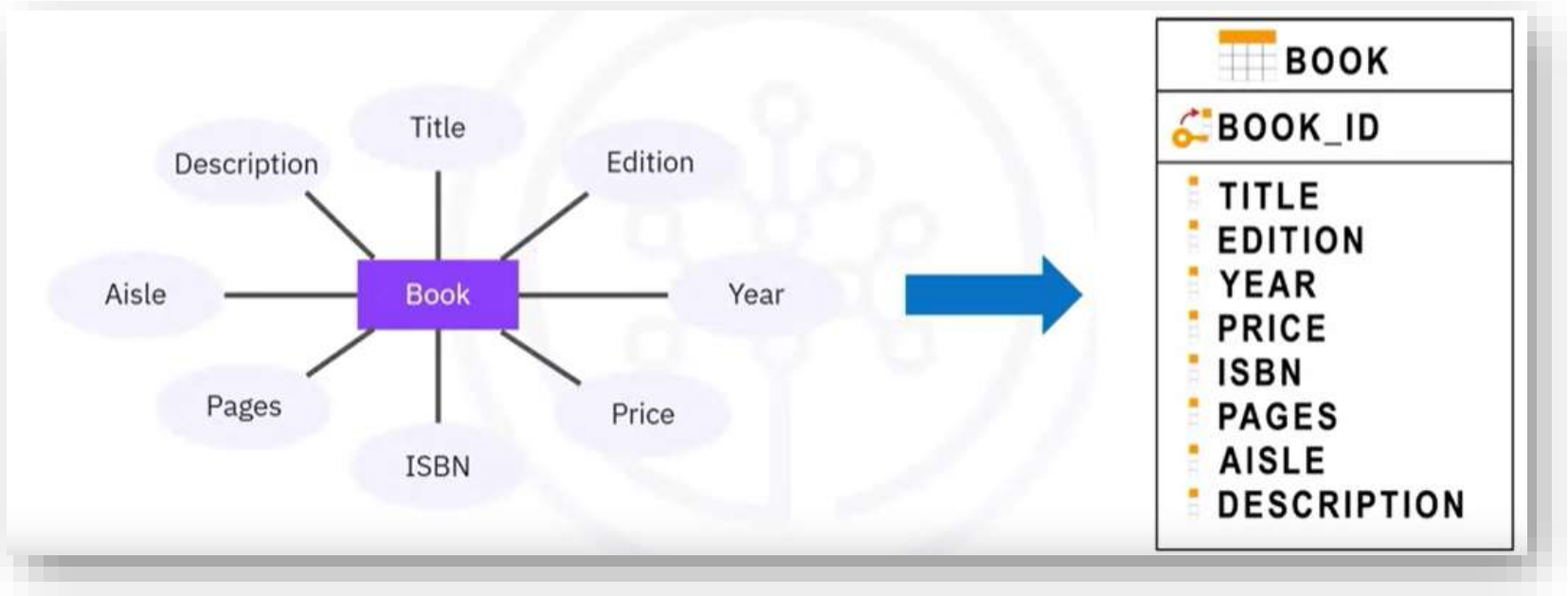
```
SAVEPOINT SAVEPOINT_NAME;
```

Relational Database Concepts

What You Will Learn

Entity-Relationship Model

- Used as a tool to design relational databases



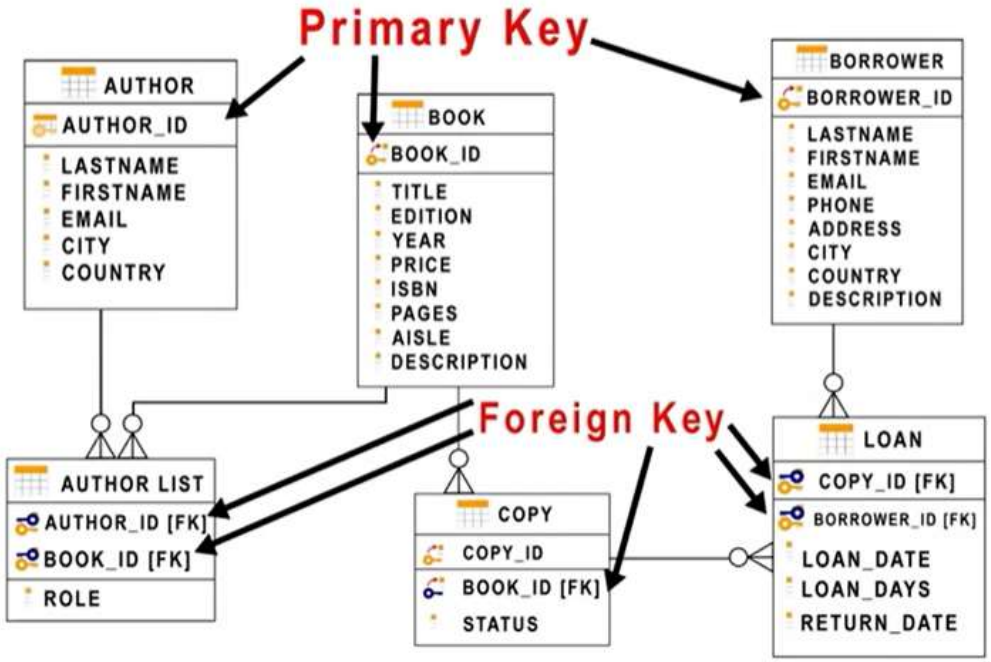
Mapping Entity Diagrams To Tables

- Entities become tables
- Attributes get translated into columns

Table: Book

Title	Edition	Year	Price	ISBN	Pages	Aisle	Description
Database Fundamentals	1	2010	24.99	978-0-98662 83-1-1	300	DB-A02	Teaches you the fundamentals of databases
Getting started with DB2 Express-C	1	2010	24.99	978-0-9866 283-5-1	280	DB-A01	Teaches you the essentials of DB2 using DB2 Express-C, the free version of DB2

Primary keys and Foreign keys



Data Definition Language (DDL)

- CREATE command
- ALTER command
- DROP command
- TRUNCATE command

CREATE Command

- Syntax

  CREATE TABLE table_name
  (column1 DATA_TYPE [CONS_TYPE CONS_NAME],
  column2 DATA_TYPE [CONS_TYPE CONS_NAME],...)

- Example

CREATE TABLE Students

(ID NUMBER(15) PRIMARY KEY, First_Name CHAR(50) NOT NULL, Last_Name CHAR(50), Address CHAR(50), City CHAR(50), Country CHAR(25), Birth_Date DATE);

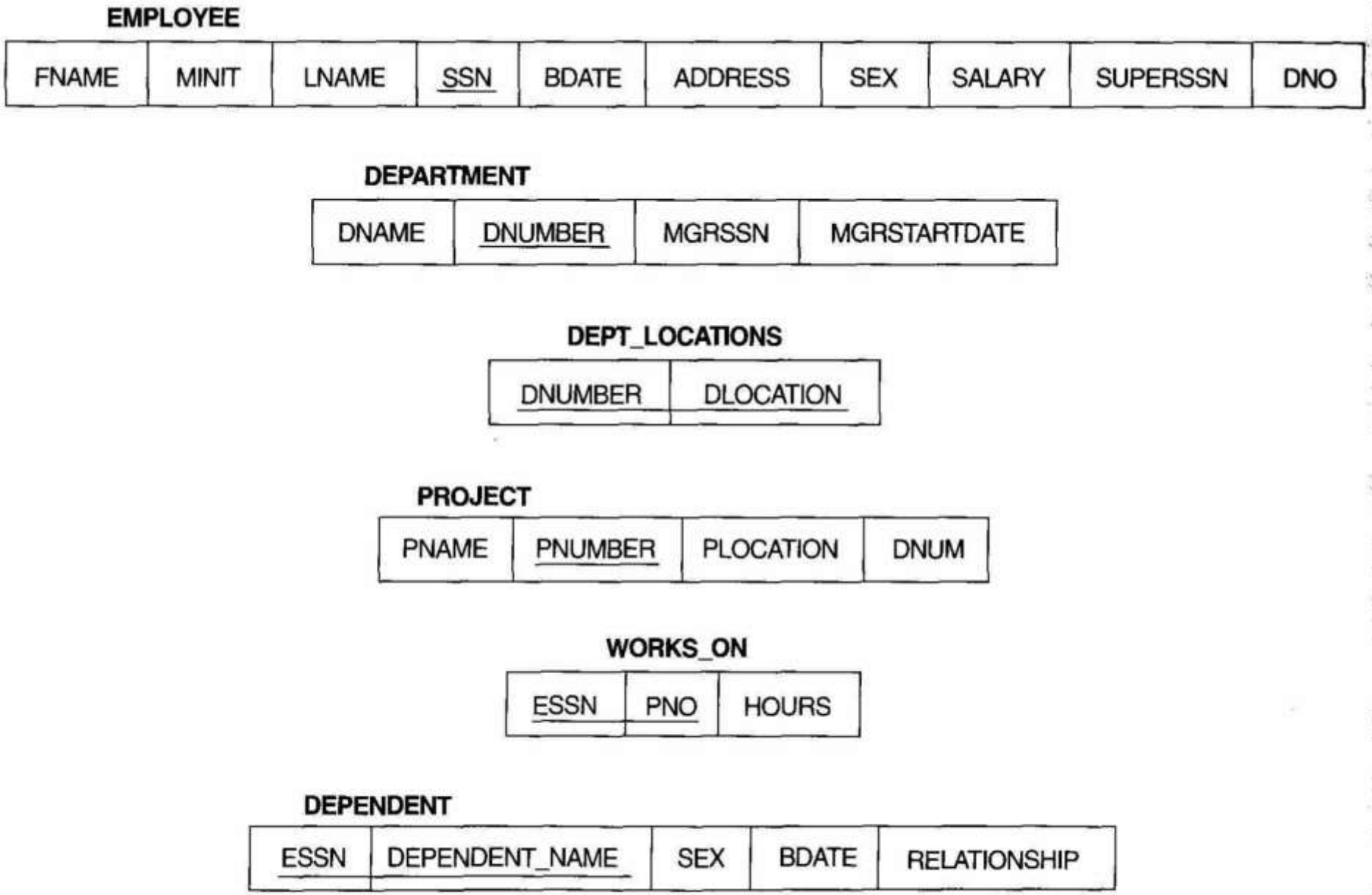


Figure 7.5 Schema diagram for the COMPANY relational database schema; the primary keys are underlined.

DROP Command

- Syntax

  DROP TABLE table_name

- Example

  DROP TABLE Students

ALTER Command

- ALTER TABLE statement is used to add or drop columns in an existing table.

- Syntax
  - ALTER TABLE table_name ADD column_name datatype
  - ALTER TABLE table_name DROP COLUMN column_name

ALTER Example

LastName	FirstName	Address
Pettersen	Kari	Storgt 20

To add a column named "City" in the "Students" table:

```
ALTER TABLE Students ADD City varchar(30)
```

Result :

LastName	FirstName	Address	City
Pettersen	Kari	Storgt 20	

Data Manipulation Language (DML)

- INSERT Command
- UPDATE Command
- DELETE Command
- SELECT Command

INSERT Command

- Syntax

```
INSERT INTO table_name (column1, column2, ...)
    VALUES (value1, value2, ...)
```

Table *Store_Information*

Column Name	Data Type
store_name	char(50)
Sales	float
Date	datetime

- Example

```
INSERT INTO Store_Information (store_name, Sales, Date)
        VALUES ('Los Angeles', 900, 'Jan-10-1999')
```

INSERT Example 2

LastName	FirstName	Address	City
El-Sayed	Mohamed	Nasr City	Cairo

```
INSERT INTO Students  VALUES ('Saleh', 'Ahmed', 'Moharam bak', 'Alex.')
```

Result

LastName	FirstName	Address	City
El-Sayed	Mohamed	Nasr City	Cairo
Saleh	Ahmed	Moharam bak	Alex.

INSERT Example 3

LastName	FirstName	Address	City
El-Sayed	Mohamed	Nasr City	Cairo
Saleh	Ahmed	Moharam bak	Alex.

```
INSERT INTO Students (LastName, City) VALUES ('Hassan', 'Assuit')
```

Result

LastName	FirstName	Address	City
El-Sayed	Mohamed	Nasr City	Cairo
Saleh	Ahmed	Moharam bak	Alex.
Hassan			Assuit

UPDATE Command

- Syntax

  UPDATE table_name

  SET column_1= new value, column_2= new value
  WHERE condition

UPDATE Example

```
UPDATE Store_Information
SET Sales = 500
WHERE store_name = 'Los Angeles'  AND Date = 'Jan-08-1999'
```

Before

store_name	*Sales*	*Date*
Los Angeles	*$1500*	*Jan-05-1999*
San Diego	*$250*	*Jan-07-1999*
Los Angeles	*$300*	*Jan-08-1999*
Boston	*$700*	*Jan-08-1999*

After

store_name	*Sales*	*Date*
Los Angeles	*$1500*	*Jan-05-1999*
San Diego	*$250*	*Jan-07-1999*
Los Angeles	*$500*	*Jan-08-1999*
Boston	*$700*	*Jan-08-1999*

UPDATE Example 2

LastName	FirstName	Address	City
El-Sayed	Mohamed	Nasr City	Cairo
Saleh	Ahmed	Moharam bak	Alex.

```
UPDATE Students SET Address = '241 El-haram', City = 'Giza'
WHERE LastName = 'El-Sayed'
```

Result :

LastName	FirstName	Address	City
El-Sayed	Mohamed	241 El-haram	Giza
Saleh	Ahmed	Moharam bak	Alex.

DELETE Command

- Syntax

  DELETE FROM table_name

  WHERE condition

DELETE Example

```
DELETE FROM  Store_Information
WHERE store_name = 'Los Angeles'
```

Before

store_name	*Sales*	*Date*
Los Angeles	*$1500*	*Jan-05-1999*
San Diego	*$250*	*Jan-07-1999*
Los Angeles	*$300*	*Jan-08-1999*
Boston	*$700*	*Jan-08-1999*

After

store_name	*Sales*	*Date*
San Diego	*$250*	*Jan-07-1999*
Boston	*$700*	*Jan-08-1999*

TRUNCATE Vs DELETE

TRUNCATE TABLE is functionally identical to DELETE statement with no WHERE clause

- TRUNCATE TABLE table_name
- TRUNCATE TABLE customer

Simple Queries

- Syntax

```
SELECT      <attribute list >
FROM        <table list>
WHERE       <condition>
ORDER BY <attribute list >
```

Examples

- SELECT *
  FROM departments;

- SELECT emp_id, emp_name, dept_id
  FROM employees;

- SELECT dept_id, dept_name
  FROM departments
  WHERE location = 'Cairo';

DISTINCT Keyword

- It's a **row** keyword that displays unique rows
- Example

Employees table

EmpNo	Name	DNo	JobID
100	Ahmed	2	Sales_Rep
200	Mai	2	IT_PROG
300	Ali	2	Sales_Rep
400	Mahmoud	3	Sales_Rep

- SELECT DISTINCT DNo
  FROM employees;

Output

DNo
2
3

DISTINCT Keyword (cont.)

- Example

Employees table

EmpNo	Name	DNo	JobID
100	Ahmed	2	Sales_Rep
200	Mai	2	IT_PROG
300	Ali	2	Sales_Rep
400	Mahmoud	3	Sales_Rep

- SELECT DISTINCT DNo,JobID
  FROM employees;

Output

DNo	JobID
2	Sales_Rep
2	IT_PROG
3	Sales_Rep

وزارة الاتصالات
وتكنولوجيا المعلومات

Questions?

رواد مصر الرقمية