

**Data Technician**

|  |
| --- |
|  |

|  |
| --- |
| Name: Nelesh Parmar |
| Course Date: 28 April 2025 |
|  |

**Table of contents**

[Day 1: Task 1 3](#_Toc106466110)

[Day 1: Task 2 3](#_Toc1673105856)

[Day 3: Task 1 4](#_Toc1682646377)

[Day 4: Task 1: Written 6](#_Toc135095808)

[Day 4: Task 2: SQL Practical 9](#_Toc1228298484)

[Course Notes 19](#_Toc631049525)

[Additional Information 19](#_Toc1612679949)

# Day 1: Task 1

Please research and complete the below questions relating to key concepts of databases.

|  |  |
| --- | --- |
| What is a primary key? | Entity Identifier, ensures uniqueness and integrity.  Unique as only 1 per table and cannot be NULL.  Used to uniquely identify records.  A Students table in a database:  StudentID (Primary Key) | Name | Age | Email  --------------------------------  1001 | Alice | 22 | alice@email.com  1002 | Bob | 24 | bob@email.com  1003 | Charlie| 23 | charlie@email.com  StudentID serves as the **Primary Key** because it uniquely identifies each student. |
| How does this differ from a secondary key? | The secondary key may not be unique thus can have more than 1 which can result in duplicates and can be NULL.  Helps with optimize searches and enable efficient data retrieval.  Used for searching, indexing, or referencing  StudentID (Primary Key) | Name | Age | Email (Secondary Key)  -----------------------------------------------------------  1001 | Alice | 22 | alice@email.com  1002 | Bob | 24 | [bob@email.com](mailto:bob@email.com)  1003 | Charlie | 23 | charlie@email.com  **Primary Key:** StudentID uniquely identifies each student.  **Secondary Key:** Email can be searched or indexed, but it may not be unique. |
| How are primary and foreign keys related? | They both provide relationships between tables. |
| Provide a real-world example of a one-to-one relationship | A person has one unique driving license. |
| Provide a real-world example of a one-to-many relationship | One customer can place many orders. |
| Provide a real-world example of a many-to-many relationship | Many students can choose many courses. |

# Day 1: Task 2

Please research and complete the below questions relating to key concepts of databases.

|  |  |
| --- | --- |
| What is the difference between a relational and non-relational database? | Relational databases are structured and stores data into tables using a predefined schema (transactional data).  Non-relational are non-structured and stores data flexible formats like documents, key-value pairs, graphs, or wide-column stores. Can be used for big data and best for use cases like real-time analytics.  Relational databases can only be scaled vertically meaning the db cannot be split but you can add more power to the server only.  Non-relational databases can be scaled both vertically and horizontally meaning, you can add more power to the server, as well as splitting the db across servers. |
| What type of data would benefit off the non-relational model?  Why? | Big data would benefit the non-relational mode.  Social media data, such as posts, comments, reactions—vary in format and structure.  IoT devices generate high-frequency streaming data, which varies in structure and requires fast processing.  Real-Time Applications (Gaming & Chat) multiplayer online games and chat applications like WhatsApp or Discord. |

# Day 3: Task 1

Please research the below ‘JOIN’ types, explain what they are and provide an example of the types of data it would be used on.

|  |  |
| --- | --- |
| Self-join | Where a table is joined with itself. This is useful when you need to compare rows within the same table, typically when dealing with hierarchical or relational data.  You have 1 table: Employees.  Each employees have a managerID, referencing another employee within the same table.  Using left join, results in a new table: |
| Right join | Retrieves all records from the right table, and only the matching records from the left table. If there is no match in the left table, the result will still include rows from the right table, but the columns from the left table will contain NULL  You have 2 tables: Employees and Departments.  You want to match employees to departments. Using right join, results in a new table:    The "Marketing" department appears in the results even though no employees belong to it, so Name is NULL |
| Full join | Also known as FULL OUTER JOIN is a SQL operation that combines results from both tables and returns all records where there is a match in either table. If a record exists in only one of the tables, the missing side will contain NULL values.  You have 2 tables: Employees and Departments.  You want to match employees to departments. Using full join, results in a new table:    Both David (who has no department) and Marketing (which has no employees assigned) appear in the result. FULL JOIN ensures that all records from both tables are included, with NULL values filling in where no match exists. |
| Inner join | Combines rows from two or more tables based on a related column. It only returns rows where there is a match in both tables.  You have 2 tables: Employees and Departments.  You want to match employees to departments. Using inner join, results in a new table:    Robert is not listed because his DepartmentID is NULL and does not match any row in Departments |
| Cross join | It combines every row from the first table with every row from the second table, resulting in a large number of combinations.  You have 2 tables: Employees and Departments.  You want to match employees to departments. Using cross join, results in a new table:    Each employee is matched with every department, leading to 3 × 2 = 6 total rows. |
| Left join | Retrieves all records from the left table, and only the matching records from the right table. If there is no match in the right table, the result will still include rows from the left table, but the columns from the right table will contain NULL values.  You have 2 tables: Employees and Departments.  You want to match employees to departments. Using left join, results in a new table:    David is included in the results, even though his DepartmentID is NULL. However, because there's no match in the Departments table, his DepartmentName is NULL. |

# Day 4: Task 1: Written

In your groups, discuss and complete the below activity. You can either nominate one writer or split the elements between you. Everyone however must have the completed work below:

*Imagine you have been hired by a small retail business that wants to streamline its operations by creating a new database system. This database will be used to manage inventory, sales, and customer information. The business is a small corner shop that sells a range of groceries and domestic products. It might help to picture your local convenience store and think of what they sell. They also have a loyalty program, which you will need to consider when deciding what tables to create.*

*Write a 500-word essay explaining the steps you would take to set up and create this database. Your essay should cover the following points:*

1. ***Understanding the Business Requirements****:*
   1. *What kind of data will the database need to store?*
   2. *Who will be the users of the database, and what will they need to accomplish?*
2. ***Designing the Database Schema****:*
   1. *How would you structure the database tables to efficiently store inventory, sales, and customer information?*
   2. *What relationships between tables are necessary (e.g., how sales relate to inventory and customers)?*
3. ***Implementing the Database****:*
   1. *What SQL commands would you use to create the database and its tables?*
   2. *Provide examples of SQL statements for creating tables and defining relationships between them.*
4. ***Populating the Database****:*
   1. *How would you input initial data into the database? Give examples of SQL INSERT statements.*
5. ***Maintaining the Database****:*
   1. *What measures would you take to ensure the database remains accurate and up to date?*
   2. *How would you handle backups and data security?*

*Your essay should include specific examples of SQL commands and explain why each step is necessary for creating a functional and efficient database for the retail business.*

|  |  |
| --- | --- |
| Please write your 500-word essay here | 1. Understanding the Business Requirements  a. Data to Store:  The database needs to store various types of information, such as:  Inventory data: product name, product category, quantity, supplier, price.  Sales data: transaction date, items sold, quantities, revenue, payment method.  Customer data: name, contact details, loyalty program ID, points earned.  b. Users and Their Needs:  Store owners/managers: who need to view inventory levels, analyze sales, and manage customers.  Cashiers: who need to record sales and update inventory.  IT or admin personnel: who maintain the database, perform backups, and ensure security.  2. Designing the Database Schema  a. Structure:  The core tables will be made up of:  Products: stores all product details.  Customers: stores customer info and loyalty points.  Sales: records each sale.  Sale item: details of products sold in each sale (since one sale can have multiple items).  Loyalty Program: defines the structure of the loyalty scheme.  b. Relationships:  One customer can have many sales.  One sale can include many sale items.  Each sale item refers to one Product.  Each product can appear in many sale items.  3. Implementing the Database  We need to create the tables by using the following SQL commands:  a. CREATE TABLE Products (  Product\_ID INT PRIMARY KEY,  ProductName VARCHAR(100),  ProductCategory VARCHAR(50),  Quantity INT,  Supplier VARCHAR(100),  Price DECIMAL(10,2)  );  b. CREATE TABLE Customers (  Customer\_ID INT PRIMARY KEY,  Name VARCHAR(100),  Email VARCHAR(100),  LoyaltyPoints INT,  Earned INT  );  c. CREATE TABLE Sales (  Sale\_ID INT PRIMARY KEY,  Customer\_ID INT,  Product\_ID INT,  Date DATETIME,  Quantity INT,  PaymentMethod VARCHAR(100),  TotalAmount DECIMAL(10,2),  FOREIGN KEY (Customer\_ID) REFERENCES Customers(Customer\_ID),  FOREIGN KEY (Product\_ID) REFERENCES Products(Product\_ID),  );  4. Populating the Database  Example INSERT Statements:  INSERT INTO Products VALUES (1, 'Milk', 'Dairy', 100, ‘Everest’, 1.20);  INSERT INTO Customers VALUES (1, 'Darth Vader, 'vader@starwars.com', '555-1234', 50);  INSERT INTO Sales VALUES (1, 1, 1, '2025-05-19 10:30:00', 3, ‘Online’, 3.60);  These commands insert a product (Milk), a customer (Alice), and a sale involving three units of milk.  5. Maintaining the Database  a. Accuracy and Updates:  Regularly update stock after each sale.  Use triggers or application logic to update LoyaltyPoints and QuantityInStock.  b. Backups and Security:  Schedule daily backups using SQL tools or automated scripts.  Limit database access using user roles and permissions. |

# Day 4: Task 2: SQL Practical

In your groups, work together to answer the below questions. It may be of benefit if one of you shares your screen with the group and as a team answer / take screen shots from there.

**Setting up the database:**

1. **Download world\_db(1)**
2. **Follow each step to create your database**

**For each question I would like to see both the syntax used and the output.**

1. **Count Cities in USA:** *Scenario:* You've been tasked with conducting a demographic analysis of cities in the United States. Your first step is to determine the total number of cities within the country to provide a baseline for further analysis.

|  |
| --- |
|  |

1. **Country with Highest Life Expectancy:** *Scenario:* As part of a global health initiative, you've been assigned to identify the country with the highest life expectancy. This information will be crucial for prioritising healthcare resources and interventions.

|  |
| --- |
|  |

1. **"New Year Promotion: Featuring Cities with 'New :** *Scenario:* In anticipation of the upcoming New Year, your travel agency is gearing up for a special promotion featuring cities with names including the word 'New'. You're tasked with swiftly compiling a list of all cities from around the world. This curated selection will be essential in creating promotional materials and enticing travellers with exciting destinations to kick off the New Year in style.

|  |
| --- |
|  |

1. **Display Columns with Limit (First 10 Rows):** *Scenario:* You're tasked with providing a brief overview of the most populous cities in the world. To keep the report concise, you're instructed to list only the first 10 cities by population from the database.

|  |
| --- |
|  |

1. **Cities with Population Larger than 2,000,000:** *Scenario:* A real estate developer is interested in cities with substantial population sizes for potential investment opportunities. You're tasked with identifying cities from the database with populations exceeding 2 million to focus their research efforts.

|  |
| --- |
|  |

1. **Cities Beginning with 'Be' Prefix:** *Scenario:* A travel blogger is planning a series of articles featuring cities with unique names. You're tasked with compiling a list of cities from the database that start with the prefix 'Be' to assist in the blogger's content creation process.

|  |
| --- |
|  |

1. **Cities with Population Between 500,000-1,000,000:** *Scenario:* An urban planning committee needs to identify mid-sized cities suitable for infrastructure development projects. You're tasked with identifying cities with populations ranging between 500,000 and 1 million to inform their decision-making process.

|  |
| --- |
|  |

1. **Display Cities Sorted by Name in Ascending Order:** *Scenario:* A geography teacher is preparing a lesson on alphabetical order using city names. You're tasked with providing a sorted list of cities from the database in ascending order by name to support the lesson plan.

|  |
| --- |
|  |

1. **Most Populated City:** *Scenario:* A real estate investment firm is interested in cities with significant population densities for potential development projects. You're tasked with identifying the most populated city from the database to guide their investment decisions and strategic planning.

|  |
| --- |
|  |

1. **City Name Frequency Analysis: Supporting Geography Education** *Scenario*: In a geography class, students are learning about the distribution of city names around the world. The teacher, in preparation for a lesson on city name frequencies, wants to provide students with a list of unique city names sorted alphabetically, along with their respective counts of occurrences in the database. You're tasked with this sorted list to support the geography teacher.

|  |
| --- |
|  |

1. **City with the Lowest Population:** *Scenario:* A census bureau is conducting an analysis of urban population distribution. You're tasked with identifying the city with the lowest population from the database to provide a comprehensive overview of demographic trends.

|  |
| --- |
|  |

1. **Country with Largest Population:** *Scenario:* A global economic research institute requires data on countries with the largest populations for a comprehensive analysis. You're tasked with identifying the country with the highest population from the database to provide valuable insights into demographic trends.

|  |
| --- |
|  |

1. **Capital of Spain:** *Scenario:* A travel agency is organising tours across Europe and needs accurate information on capital cities. You're tasked with identifying the capital of Spain from the database to ensure itinerary accuracy and provide travellers with essential destination information.

|  |
| --- |
|  |

1. **Country with Shortest Life Expectancy:** *Scenario:* A healthcare foundation is conducting research on global health indicators. You're tasked with identifying the country with the highest life expectancy from the database to inform their efforts in improving healthcare systems and policies.

|  |
| --- |
|  |

1. **Cities in Europe:** *Scenario:* A European cultural exchange program is seeking to connect students with cities across the continent. You're tasked with compiling a list of cities located in Europe from the database to facilitate program planning and student engagement.

|  |
| --- |
|  |

1. **Average Population by Country:** *Scenario:* A demographic research team is conducting a comparative analysis of population distributions across countries. You're tasked with calculating the average population for each country from the database to provide valuable insights into global population trends.

|  |
| --- |
|  |

1. **Capital Cities Population Comparison:** *Scenario:* A statistical analysis firm is examining population distributions between capital cities worldwide. You're tasked with comparing the populations of capital cities from different countries to identify trends and patterns in urban demographics.

|  |
| --- |
|  |

1. **Countries with Low Population Density:** *Scenario:* An agricultural research institute is studying countries with low population densities for potential agricultural development projects. You're tasked with identifying countries with sparse populations from the database to support the institute's research efforts.

|  |
| --- |
|  |

1. **Cities with High GDP per Capita:** *Scenario:* An economic consulting firm is analysing cities with high GDP per capita for investment opportunities. You're tasked with identifying cities with above-average GDP per capita from the database to assist the firm in identifying potential investment destinations.

|  |
| --- |
|  |

1. **Display Columns with Limit (Rows 31-40):** *Scenario:* A market research firm requires detailed information on cities beyond the top rankings for a comprehensive analysis. You're tasked with providing data on cities ranked between 31st and 40th by population to ensure a thorough understanding of urban demographics.

|  |
| --- |
|  |

|  |
| --- |
| **Course Notes** |

It is recommended to take notes from the course, use the space below to do so, or use the revision guide shared with the class:

|  |
| --- |
| **Additional Information** |

We have included a range of additional links to further resources and information that you may find useful, these can be found within your revision guide.

**END OF WORKBOOK**

**Please check through your work thoroughly before submitting and update the table of contents if required.**

**Please send your completed work booklet to your trainer.**