

**Data Technician**

|  |
| --- |
|  |

|  |
| --- |
| Name: |
| Course Date: |
|  |

**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? | A **Primary Key (PK)** is a unique identifier for each record in a database table.  It must be **unique** for every row and **cannot be null**.  It ensures that each record can be uniquely identified.  **Example**: A customer\_ID in a customer table. |
| How does this differ from a secondary key? | A **Secondary Key** (also called an alternate key or candidate key) is a key used for **searching or indexing** but is **not the primary key**.  It may or may not be unique, depending on its purpose.  It helps speed up queries but doesn’t uniquely identify records like the primary key. |
| How are primary and foreign keys related? | A **Foreign Key (FK)** is an attribute (or set of attributes) in one table that **references the primary key** in another table.  It is used to **enforce referential integrity** and link tables together. |
| Provide a real-world example of a one-to-one relationship | **Example:** Each person has one passport; each passport belongs to one person.  Tables: Person and Passport linked by PersonID.  Each person row corresponds to exactly one passport row. |
| Provide a real-world example of a one-to-many relationship | A teacher teaches many students, but each student has one main teacher. |
| Provide a real-world example of a many-to-many relationship | Students enrol in many courses, and each course has many students. |

# 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? |  |
| What type of data would benefit off the non-relational model?  Why? | **Unstructured or Semi-structured data** — such as JSON documents, XML, logs, social media posts, sensor data.  **Highly variable data** where schema changes often or is not strictly defined.  **Big data and real-time analytics** — systems that need to handle massive volumes of data quickly.  **Hierarchical or graph-based data** — like social networks, recommendation engines.  **Caching and session management** — using key-value stores like Redis.  **WHY**  **Flexibility:** You don’t need to predefine a strict schema, so you can easily store varying data formats and evolve the data model without downtime.  **Scalability:** Horizontal scaling makes it easier to handle massive amounts of data distributed over many servers.  **Performance:** Optimized for specific use cases like fast reads/writes of large volumes or complex graph traversals.  **Complex data types:** Some data is naturally hierarchical or graph-based, which is cumbersome to model in relational tables. |

# 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 | **Definition**: A table joined with itself using aliases.  **Use Case**: Useful for hierarchical data or comparing rows within the same table.  **Example Data**: Employee table with columns EmployeeID, LastName, ManagerID (where ManagerID refers to another EmployeeID). |
| Right join | **Definition**: Returns all rows from the right table, plus matched rows from the left. Unmatched left‑table columns are NULL.  **Use Case**: Rarely needed if you can swap table order and use a Left Join. But useful for making sure you include all rows from the “right” side.  **Example Data**: Same Employee and Department. |
| Full join | **Definition**: Returns all rows from both tables. Matched rows are joined; unmatched rows show NULL in the missing side.  **Use Case**: When you need every record from both tables—even if there is no matching counterpart.  **Example Data**: Employee and Department. |
| Inner join | **Definition**: Returns only rows that have matching values in both tables.  **Use Case**: When you only care about intersecting data.  Example Data: Employee and Department where Employee.DepartmentID = Department.DepartmentID. |
| Cross join | **Definition**: Cartesian product of both tables—each row of one paired with every row of the other. No join condition required.  **Use Case**: When you need all possible combinations of two datasets. Often used for matrix-style enumerations or testing.  **Example Data**: Department and SeniorityLevels (e.g. junior, mid, senior). |
| Left join | **Definition**: Returns all rows from the left table, plus matched rows from the right. Unmatched right‑table columns are NULL.  **Use Case**: When you want all primary rows and supplementary data if available.  **Example Data**: Employee and Department. |

# 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 | Day 4 – Task 1:  To streamline operations for a small retail business, such as a local convenience store, creating a well-structured database system is essential. This database must manage inventory, sales transactions, customer data, and a loyalty program. The process involves several key steps: understanding business needs, designing the schema, implementing the database, populating it with data, and ensuring ongoing maintenance.  **Understanding the Business Requirements** First, it's important to identify what kind of data the business needs to store. The database must track product details (e.g., name, category, price, stock quantity), sales transactions (date, items sold, total amount), customer information (name, contact, loyalty points), and users (employees who interact with the system). The main users of this database will be store staff and managers. Staff need to process sales and check inventory, while managers require reports on sales performance and customer activity.  **Designing the Database Schema** The next step is designing an efficient schema. The following tables are necessary:   * **Products**: stores item details. * **Customers**: stores customer contact info and loyalty points. * **Sales**: records each sale and links to a customer. * **SaleItems**: records which products were sold in each transaction. * **Users**: tracks who is using the system.   Key relationships include: one customer can have many sales, each sale can include multiple items, and each item sold must link to a product. This relational structure allows for accurate data retrieval and integrity.  **Implementing the Database** SQL is used to create the database and its tables. For example:  1. CREATE TABLE Products (  2. ProductID INT PRIMARY KEY AUTO\_INCREMENT,  3. Name VARCHAR(100),  4. Category VARCHAR(50),  5. Price DECIMAL(10,2),  6. QuantityInStock INT  7. );  8.  9. CREATE TABLE Customers (  10. CustomerID INT PRIMARY KEY AUTO\_INCREMENT,  11. Name VARCHAR(100),  12. Email VARCHAR(100),  13. Phone VARCHAR(20),  14. LoyaltyPoints INT DEFAULT 0  15. );  16.  17. CREATE TABLE Sales (  18. SaleID INT PRIMARY KEY AUTO\_INCREMENT,  19. CustomerID INT,  20. SaleDate DATE,  21. TotalAmount DECIMAL(10,2),  22. FOREIGN KEY (CustomerID) REFERENCES Customers(CustomerID)  23. );  24.  25. CREATE TABLE SaleItems (  26. SaleItemID INT PRIMARY KEY AUTO\_INCREMENT,  27. SaleID INT,  28. ProductID INT,  29. Quantity INT,  30. UnitPrice DECIMAL(10,2),  31. FOREIGN KEY (SaleID) REFERENCES Sales(SaleID),  32. FOREIGN KEY (ProductID) REFERENCES Products(ProductID)  33. );  34.  35. INSERT INTO Products (Name, Category, Price, QuantityInStock)  36. VALUES ('Milk', 'Dairy', 1.20, 50);  37.  38. INSERT INTO Customers (Name, Email, Phone)  39. VALUES ('Jane Doe', 'jane@example.com', '555-1234');  **Maintaining the Database** To keep data accurate, application logic or database triggers can manage stock levels and loyalty points. Regular audits, constraints, and validation rules help maintain integrity. For backups and security, schedule automatic backups and use role-based access control. By following these steps, the business can manage operations efficiently, ensuring reliable access to inventory, sales, and customer data. |

# 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.**