

**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 is a column in a database table that:   1. Uniquely identifies each row in the table. 2. Cannot have a NULL value. 3. Must be unique across all rows. |
| How does this differ from a secondary key? | * key used for searching or indexing data that is not the primary key where primary key is the main identifier of the row. * Secondary key may be unique but not mandatory where as primary key must be unique. * It may allow NULL values but primary key don’t allow NULL values |
| How are primary and foreign keys related? | Foreign key is a column (or set of columns) in one table that refers to the primary key in another table. It establishes relationship between the tables. It is a child table’s link to parent(Primary key table). |
| Provide a real-world example of a one-to-one relationship | Human Body and DNA Profile   * Each person has exactly one unique DNA profile. * Each DNA profile belongs to exactly one person. |
| Provide a real-world example of a one-to-many relationship | Music Album and Songs   * One album can contain many songs. * Each song belongs to exactly one album. |
| Provide a real-world example of a many-to-many relationship | Athletes and Sports Events   * One athlete can participate in multiple events. * One event can have many athletes. |

# 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? | | **Feature** | **Relational (RDBMS)** | **Non-Relational (NoSQL)** | | --- | --- | --- | | Data Structure | Tables (rows & columns) | Documents, key-value, graph, wide-column | | Schema | Fixed (strict) | Flexible or schema-less | | Query Language | SQL | Varies (MongoDB queries, APIs) | | Relationships | Built-in (PK, FK) | Often handled in app or embedded | | Scalability | Vertical (scale-up) | Horizontal (scale-out) | | Integrity | Strong (ACID transactions) | Often eventual consistency (BASE) | |
| What type of data would benefit off the non-relational model?  Why? | **Types of Data Best Suited for Non-Relational (NoSQL) Databases:**   1. Unstructured Data 2. Semi-Structured Data 3. Rapidly Changing / Dynamic Data 4. Big Data 5. Highly Connected Data   **Why These Data Benefit from Non-Relational Models:**   * **Schema Flexibility:** No need to redesign the database when new fields/attributes appear. * **Scalability:** Can handle massive amounts of data by scalinghorizontally (adding servers). * **Speed:** Optimized for fast reads/writes, even under heavy loads. * **Variety:** Can store different types of data (JSON, key-value, graphs, documents) in the same database. * **Distributed Nature:** Perfect for global apps with millions of users. |

# 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 | A Self JOIN is when a table is joined with itself.   * It is used when rows in the same table are related to each other. * Treat the same table as if it were two separate tables by giving each instance an alias.   Self-join is used in:  Hierarchical data: Employees who report to managers in the same table.  Comparisons within a table: For example, finding products that cost more than other products.  Matching relationships: Students assigned as mentors to other students. |
| Right join | A RIGHT JOIN returns:   * All rows from the right table, and * The matching rows from the left table. * If there’s no match, the result shows NULL for the left table’s columns.   Useful in situations like:   * Customers vs Orders (to see which customers placed orders, and which have none). * Students vs Enrolments (which students are enrolled, which are not). |
| Full join | A FULL JOIN combines the results of a LEFT JOIN and a RIGHT JOIN:   * Returns all rows from both tables. * Where there’s a match → shows combined data. * Where there’s no match → shows NULL for the missing side.   Full Join is useful :  To find all records across two tables, regardless of whether they match.  Useful for:   * Comparing datasets (e.g., old vs new customers). * Merging data from different sources. * Audit reports (e.g., orders that don’t have customers, and customers that don’t have orders). |
| Inner join | An INNER JOIN returns only the rows where there is a match in both tables.  Rows that don’t have matching values in the join condition are excluded.  Inner Join is useful:  When we want records that exist in both tables.  Common scenarios:   * Customers who placed orders. * Students enrolled in courses. * Employees assigned to departments. |
| Cross join | A CROSS JOIN returns the Cartesian product of two tables.  That means every row in the first table is combined with every row in the second table.  Unlike INNER JOIN or LEFT JOIN, it does not need a join condition  Cross Join is useful:  Generating combinations of two datasets.  Examples:   * Creating a schedule (all employees × all shifts). * Matching products × sizes for an inventory list. * Creating test data when you need every possible pairing. |
| Left join | A LEFT JOIN (or LEFT OUTER JOIN) returns:   * All rows from the left table, and * The matching rows from the right table.   If there’s no match in the right table, the result shows NULL for the right table’s columns.   * To see all records from a primary table and include related information if it exists.   Left Join is useful in:   * Common scenarios:   + Customers and Orders → show all customers and their orders (even if they haven’t ordered anything).   + Students and Enrolments → list all students and their course enrolments.   + Products and Sales → list all products and sales data, including unsold products. |

# 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:   The first step is gathering the business requirements. The database must store data like:   * products (e.g., product name, category, price, and stock quantity) * sales transactions (e.g., date, items sold, total amount) * customers (e.g., name, contact details, loyalty points).   The users of the database will mainly be shop staff and the manager. Staff will use it to process sales, check stock levels, loyalty points and update product information. The manager may use it to run reports on sales trends, customer activity, and inventory levels. |

# 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.

|  |
| --- |
| select count(c.ID) As Tot\_count, cu.Name from city c inner join country cu on c.CountryCode= cu.Code  group by cu.Name having cu.Name="United States"; |

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.**