|  |
| --- |
|  |

|  |
| --- |
| Name: Tasleem Akhter |
| Course Date: 12/05/25 |
|  |

**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 unique identifier for each record in a database table. It has no duplicate values or empty entries and is a fundamental concept in database design. |
| How does this differ from a secondary key? | The primary key differs from secondary key because the secondary key is used for searching or sorting and can be multiple, may not be unique, also be indexed manually to improve performance. |
| How are primary and foreign keys related? | Primary and foreign keys are related through linking their roles in linking tables in a relational database. And also establish a relationship between tables, data integrity and help with queries across related data. |
| Provide a real-world example of a one-to-one relationship | A real world example can be for example every person has a unique passport and only that person can use their passport and no-one else because that is a one-to-one relationship. |
| Provide a real-world example of a one-to-many relationship | Real world example of a one-to-one many relationship can be that one tutor can teach many students. For example, in a database, the teacher table has the primary key and the student table has the foreign key to in order to link each student to their table. |
| Provide a real-world example of a one-to-many relationship | Real world example of one-to-many relationships can include a library which can consist of many books. In a database structure this could be library table as the primary key and the book key as a foreign key to link each book to the library. |

# 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 (SQL) uses structured tables with fixed schemes which are ideal for complex queries and examples and also best for structured data like financial records. The non-relational (NoSQL) uses flexible formats like key value pairs and are schema-less and scalable which is useful for unstructured or rapidly changing data like social media posts. |
| What type of data would benefit off the non-relational model?  Why? | Data that is unstructured or changing data (like social media, logs, multimedia), nested or varied formats like user profiles, high-volume, fast-moving data like IoT or analytics. The reason for this is that all these systems need flexible, scalable storage because the data does not fit neatly into rows and columns. |

# 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 to itself. You use aliases to treat it as two separate tables. An example is employees table where each employee has a ManagerID that refers to another employee in the same table. A self-join lets you list employees alongside their managers. This type can be used for category-subcategory relationships and comparing rows within the same table. |
| Right join | A right join is returns all records from the **right table**  and the matched records from the . If there left tables no match, NULLs appear for the left table. Its uses are making sure that all the entries from the right table are included and reporting completeness of data. |
| Full join | A full join returns all records when there is a match in either left or right table. Unmatched rows from both sides are included with NULLs. It is useful for identifying gaps or mismatches between two sources and combining data sources with partial overlap. |
| Inner join | An inner join is a type of SQL join that returns only the rows where there is a match in both table based on a shared column. It combines data from two tables by comparing values in a specified column like a foreign key in one table and a primary key in another. It is useful for filtering matched records to make results cleaner and more relevant and also data analysis which is useful when you need to combine datasets for reporting or insights |
| Cross join | A cross join in SQL is like throwing two decks of cards on the table and pairing every card from one deck with every card from the other. This produces the cartesian product of two table which means that every row from the first table is combined with every row from the second regardless of any matching condition. This can generate all combinations of data like; all possible meal and drink combos from two menus or matching every employee with every training program. It can also be used for testing and simulation. |
| Left join | A left join in SQL is used to combine data from two tables which is returning all the rows from the left table and matching rows from the right table. If there’s no match, the result will include NULL values for the right table’s columns. Common uses for left join (SQL) include data auditing like to identify missing relationships such as products without sales. And also default values which combine with COALESCE() to fill in blanks for unmatched rows of data. Also useful for showing all records from one table, even if related data is missing. |

# 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 | To set up a database system for a small retail business, the first step is understanding its operational needs. The database must store data on inventory (product names, categories, quantities, prices), sales (transaction dates, items sold, quantities, totals), customers (names, contact details, loyalty status), and loyalty program details (points earned, redemption history). Users of the database will include shop staff, managers, and possibly external auditors. Staff will need to track stock levels, process sales, and manage customer accounts, while managers will require insights into sales trends and customer engagement.  Designing the database schema involves structuring tables to reflect these needs. A normalized schema would include the following tables: Products, Customers, Sales, and Loyalty. The Products table stores item details. Customers holds personal and loyalty data. Sales records each transaction, while Sale items links products to sales, allowing multiple items per transaction. Loyalty tracks points earned and redeemed. Relationships are crucial: Sales must link to Customers, and Sale items must link to both Sales and Products.  To implement the database, SQL commands are used to create tables and define relationships. For example:  CREATE TABLE Products ( ProductID INT PRIMARY KEY, Name VARCHAR(100), Category VARCHAR(50), Price DECIMAL(5,2), StockQuantity INT ); CREATE TABLE Customers ( CustomerID INT PRIMARY KEY, Name VARCHAR(100), Email VARCHAR(100), LoyaltyPoints INT DEFAULT 0 ); CREATE TABLE Sales ( SaleID INT PRIMARY KEY, CustomerID INT, SaleDate DATE, FOREIGN KEY (CustomerID) REFERENCES Customers(CustomerID) ); CREATE TABLE SaleItems ( SaleItemID INT PRIMARY KEY, SaleID INT, ProductID INT, Quantity INT, FOREIGN KEY (SaleID) REFERENCES Sales(SaleID), FOREIGN KEY (ProductID) REFERENCES Products(ProductID) );  Then populating the database begins with inserting initial data. For example:  INSERT INTO Products VALUES (1, 'Milk', 'Dairy', 1.20, 50); INSERT INTO Customers VALUES (1, 'Jane Doe', 'jane@example.com', 100); INSERT INTO Sales VALUES (1, 1, '2025-08-29'); INSERT INTO SaleItems VALUES (1, 1, 1, 2);  These statements add a product, a customer, a sale, and the items sold in that sale.  Maintaining the database requires regular updates to reflect stock changes, new customers, and sales. Triggers or stored procedures can automate stock deductions after sales. Data validation rules help prevent errors, such as negative stock levels. Backups should be scheduled daily, stored securely offsite or in the cloud. Access control is essential—only authorized users should modify data, and sensitive customer information must be encrypted.  In conclusion, a well-designed and maintained database streamlines operations, improves accuracy, and supports business growth. By carefully planning the schema, implementing robust SQL structures, and enforcing data integrity and security, the corner shop can efficiently manage its inventory, sales, and customer relationships to ensure data remains secure and up-to-date |

# Day 4: Task 2: SQL Practical

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.

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:**

|  |
| --- |
| SELECT COUNT(\*) AS total\_cities\_in\_usa  FROM city  WHERE countrycode = 'USA';  ( I am not very sure about how to go about this task) |

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.**