Just IT

B2W group

Apprenticeships | Training | Recruitment

Data Technician

Name:
Course Date:

Table of contents

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 uniquely labels each record in a table, the table can only have one primary key, which can have one or many columns. No null values are allowed.
How does this differ from a secondary key?	Any keys after the primary key are secondary keys. They can be indexed to access data efficiently. There can be multiple secondary keys in a table.
How are primary and foreign keys related?	Foreign key is at least a column in one table that references with the primary key in another table. This creates a link between the two tables. They are related due to referential integrity, so value in foreign key column must be null or exist in primary key column.
Provide a real-world example of a one-to-one relationship	Individuals and their passports. Each individual has one passport and vice versa.
Provide a real-world example of a one-to-many relationship	Department and employees. One department such as HR can have multiple employees, but each employee only belongs to one department.
Provide a real-world example of a many-to-many relationship	Students and classes. As students can take multiple classes, and each class can have multiple 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?	Relational database sorts data into separate tables, where columns contain data attributes and the rows contain data values. Can then link tables to get relationship. Non-relational database uses other data models to access and manage data, such as documents and keynotes. Relational database has vertical scaling, increasing/ decreasing capacity workload. While non-relational database has horizontal scaling, spreading workload across devices. Non relational database therefore more flexible due to less structure than relational database.
What type of data would benefit off the non-relational model? **Why?**	Unstructured or semi-structured data such as images and articles. Non-relational database ideal for data that doesn't fit into tables. Large data would benefit non-relational model due to horizontal scaling, as can distribute large data across multiple servers, spreading the workload. This processes data faster, making them ideal for applications that require real time data input, such as social media feeds.

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	Self-join is regular join, but the table is joined with itself.

Right join	RIGHT JOIN returns all records from the right table, and matching records from left table if there are any. (Nulls in left) even if there are no matches in the left tables. E.g.
Full join	Returns connected rows and unconnected rows from both left and right table
Inner join	INNER JOIN selects records that have matching values in both tables, (matching information from left and right table). The following SQL statement selects all products with their suppliers: SELECT SupplierName, ProductName FROM Products INNER JOIN Suppliers on Products.SupplierID = Suppliers.SupplierID
Cross join	CROSS JOIN returns all matching records from both tables, even if other table matches or not. So, rows with no matches will be listed as well. Can return very large result sets.
Left join	LEFT JOIN returns all records from the left table, even if there are no matches in the right table. (Nulls in right). E.g.

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

The database needs to store several types of information such as:

- Inventory data: product names, categories, prices, quantities, suppliers.
- Sales data: dates, times, products sold, quantities, prices, and payment methods.
- Customer data: names, contact information, and loyalty points.
- Loyalty program: points earned per purchase, total points, and redemption history.

The users of the database include the shop owner, employees, and maybe an accountant. Employees will need to update inventory and process sales, while the owner might analyse sales trends and monitor loyalty rewards.

Designing the Database Schema, on SQL

To organise the data, we could use the following tables:

1. Products (ProductID, ProductName, Category, Price, QuantityInStock)
2. Customers (CustomerID, Name, Phone, Email, LoyaltyPoints)
3. Sales (SaleID, CustomerID, SaleDate, TotalAmount)
4. SaleDetails (SaleDetailID, SaleID, ProductID, QuantitySold, PriceAtSale)
5. Suppliers (SupplierID, SupplierName, ContactInfo)
6. ProductSuppliers (ProductID, SupplierID)

Relationships:

- Each Sale links to one Customer.
- Each Sale includes multiple SaleDetails, which reference Products.
- Products can have multiple Suppliers, hence a many-to-many relationship.
-

Using SQL, to implement database, we would create the following tables:

```
CREATE TABLE Customers (
    CustomerID INT PRIMARY KEY AUTO_INCREMENT,
    Name VARCHAR(100),
    Phone VARCHAR(20),
    Email VARCHAR(100),
    LoyaltyPoints INT DEFAULT 0
);
```

```
CREATE TABLE Products (
    ProductID INT PRIMARY KEY AUTO_INCREMENT,
    ProductName VARCHAR(100),
    Category VARCHAR(50),
    Price DECIMAL(10,2),
    QuantityInStock INT
);

CREATE TABLE Sales (
    SaleID INT PRIMARY KEY AUTO_INCREMENT,
    CustomerID INT,
    SaleDate DATETIME,
    TotalAmount DECIMAL(10,2),
    FOREIGN KEY (CustomerID) REFERENCES Customers(CustomerID)
);

CREATE TABLE SaleDetails (
    SaleDetailID INT PRIMARY KEY AUTO_INCREMENT,
    SaleID INT,
    ProductID INT,
    QuantitySold INT,
    PriceAtSale DECIMAL(10,2),
    FOREIGN KEY (SaleID) REFERENCES Sales(SaleID),
    FOREIGN KEY (ProductID) REFERENCES Products(ProductID)
);
```

Then, we insert data using SQL INSERT statements:

```
INSERT INTO Products (ProductName, Category, Price, QuantityInStock)
VALUES ('Milk', 'Dairy', 2.25, 60);

INSERT INTO Customers (Name, Phone, Email)
VALUES ('John Smith', '07123456789', 'john@example.com');

INSERT INTO Sales (CustomerID, SaleDate, TotalAmount)
VALUES (1, NOW(), 4.70);

INSERT INTO SaleDetails (SaleID, ProductID, QuantitySold, PriceAtSale)
VALUES (1, 1, 5, 1.30);
```

To maintain accuracy of data:

- Regular inventory updates after sales.
- Constraints and triggers to prevent negative stock levels.
- Scheduled backups to external storage to protect against data loss.
- Access control to restrict sensitive information.

Security measures include using user roles and encrypted connections to prevent unauthorised access and secure personal information, complying with laws and regulations.

In conclusion, making this database involves understanding the shop's operational needs, designing a normalised schema, implementing it with SQL, and ensuring long-term reliability through updates, backups, and access control. This structured approach supports smooth business operations and enhances customer engagement through effective loyalty tracking.

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) here**
2. **Follow each step to create your database here**

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(*) AS TotalCitiesInUSA
FROM City
WHERE CountryCode = 'USA';
```

Answer got was 274 cities in USA.

2. **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.

```
SELECT Name AS Country, LifeExpectancy
FROM Country
WHERE LifeExpectancy IS NOT NULL
ORDER BY LifeExpectancy DESC
LIMIT 1;
```

Answer got was Andorra with 83.5 (years).

3. **"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.

```
SELECT Name AS City, CountryCode
FROM City
WHERE Name LIKE ‘%New%';

Answers:

City, CountryCode
'Newcastle', 'AUS'
'Newcastle upon Tyne', 'GBR'
'Newport', 'GBR'
'Newcastle', 'ZAF'
'Kowloon and New Kowloon', 'HKG'
'New Bombay', 'IND'
'New Delhi', 'IND'
'Khanewal', 'PAK'
'New York', 'USA'
'New Orleans', 'USA'
'Newark', 'USA'
'Newport News', 'USA'
'New Haven', 'USA'
'New Bedford', 'USA'
```

4. **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.

```
SELECT Name AS City, CountryCode, Population
FROM City
ORDER BY Population DESC
LIMIT 10; City, CountryCode, Population

Answers:
'Mumbai (Bombay)', 'IND', '10500000'
'Seoul', 'KOR', '9981619'
'São Paulo', 'BRA', '9968485'
'Shanghai', 'CHN', '9696300'
'Jakarta', 'IDN', '9604900'
'Karachi', 'PAK', '9269265'
'Istanbul', 'TUR', '8787958'
'Ciudad de México', 'MEX', '8591309'
'Moscow', 'RUS', '8389200'
'New York', 'USA', '8008278'
```

5. **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.

```
SELECT Name AS City, CountryCode, Population
FROM City
WHERE Population > 2000000
ORDER BY Population DESC;
```

Answers, 92 rows were returned, these are some of them:

```
 City, CountryCode, Population
'Mumbai (Bombay)', 'IND', '10500000'
'Seoul', 'KOR', '9981619'
'São Paulo', 'BRA', '9968485'
'Shanghai', 'CHN', '9696300'
'Jakarta', 'IDN', '9604900'
'Karachi', 'PAK', '9269265'
'Istanbul', 'TUR', '8787958'
'Ciudad de México', 'MEX', '8591309'
'Moscow', 'RUS', '8389200'
'New York', 'USA', '8008278'
'Tokyo', 'JPN', '7980230'
'Peking', 'CHN', '7472000'
'London', 'GBR', '7285000'
'Delhi', 'IND', '7206704'
'Cairo', 'EGY', '6789479'
'Teheran', 'IRN', '6758845'
'Lima', 'PER', '6464693'
```

6. **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.

```
SELECT Name AS City, CountryCode, Population
FROM City
WHERE Name LIKE 'Be%'
ORDER BY Name ASC;
```

Answers, 51 rows returned, these are some of them:

City, CountryCode, Population
'Beau Bassin-Rose Hill', 'MUS', '100616'
'Beaumont', 'USA', '113866'
'Beawar', 'IND', '105363'
'Béchar', 'DZA', '107311'
'Beerseba', 'ISR', '163700'
'Bei´an', 'CHN', '204899'
'Beihai', 'CHN', '112673'
'Beipiao', 'CHN', '194301'
'Beira', 'MOZ', '397368'
'Beirut', 'LBN', '1100000'
'Béjaïa', 'DZA', '117162'
'Bekasi', 'IDN', '644300'
'Belém', 'BRA', '1186926'
'Belfast', 'GBR', '287500'
'Belford Roxo', 'BRA', '425194'
'Belgaum', 'IND', '326399'
'Belgorod', 'RUS', '342000'
'Belize City', 'BLZ', '55810'
'Bellary', 'IND', '245391'
'Bellevue', 'USA', '109569'
'Bello', 'COL', '333470'
'Belmopan', 'BLZ', '7105'
'Belo Horizonte', 'BRA', '2139125'
'Bender (Tîghina)', 'MDA', '125700'
'Bene Beraq', 'ISR', '133900'
'Bengasi', 'LBY', '804000'
'Bengbu', 'CHN', '449245'

7. **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.

```
SELECT Name AS City, CountryCode, Population
FROM City
WHERE Population BETWEEN 500000 AND 1000000
ORDER BY Population DESC;
```

Answer, 303 rows returned, these are some:

City, CountryCode, Population
'Amman', 'JOR', '1000000'
'Mogadishu', 'SOM', '997000'
'Volgograd', 'RUS', '993400'
'Sendai', 'JPN', '989975'
'Peshawar', 'PAK', '988005'
'Baotou', 'CHN', '980000'
'Adelaide', 'AUS', '978100'
'Madurai', 'IND', '977856'
'Mekka', 'SAU', '965700'
'Köln', 'DEU', '962507'
'Managua', 'NIC', '959000'
'Detroit', 'USA', '951270'
'Shenzhen', 'CHN', '950500'
'Haora (Howrah)', 'IND', '950435'
'Campinas', 'BRA', '950043'
'Brazzaville', 'COG', '950000'
'Khartum', 'SDN', '947483'
'Karaj', 'IRN', '940968'
'Taichung', 'TWN', '940589'
'Santa Cruz de la Sierra', 'BOL', '935361'
'Varanasi (Benares)', 'IND', '929270'
'Patna', 'IND', '917243'
'Hohhot', 'CHN', '916700'
'Rosario', 'ARG', '907718'
'Voronez', 'RUS', '907700'
'Soweto', 'ZAF', '904165'
'Torino', 'ITA', '903705'

8. **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.

```
SELECT Name AS City, CountryCode, Population
FROM City
ORDER BY Name ASC;
```

Answer, 1000 rows returned, these are some:

```
 City, CountryCode, Population
'[San Cristóbal de] la Laguna', 'ESP', '127945'
'´s-Hertogenbosch', 'NLD', '129170'
'A Coruña (La Coruña)', 'ESP', '243402'
'Aachen', 'DEU', '243825'
'Aalborg', 'DNK', '161161'
'Aba', 'NGA', '298900'
'Abadan', 'IRN', '206073'
'Abaetetuba', 'BRA', '111258'
'Abakan', 'RUS', '169200'
'Abbotsford', 'CAN', '105403'
```

9. **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.

```
SELECT Name AS City, CountryCode, Population
FROM City
ORDER BY Population DESC
LIMIT 1;
```

Answer:

```
City, CountryCode, Population
'Mumbai (Bombay)', 'IND', '10500000'
```

10. **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.

```
SELECT Name AS City, COUNT(*) AS Frequency
FROM City
GROUP BY Name
ORDER BY Name ASC;
```

Answer, 1000 rows returned, these are some:

City, Frequency
'[San Cristóbal de] la Laguna', '1'
'´s-Hertogenbosch', '1'
'A Coruña (La Coruña)', '1'
'Aachen', '1'
'Aalborg', '1'
'Aba', '1'
'Abadan', '1'
'Abaetetuba', '1'
'Abakan', '1'
'Abbotsford', '1'

11. **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.

```
SELECT Name AS City, CountryCode, Population
FROM City
ORDER BY Population ASC
LIMIT 1;
```

Answer:

City, CountryCode, Population
'Adamstown', 'PCN', '42'

12. **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.

```
SELECT Name AS Country, Population
FROM Country
ORDER BY Population DESC
LIMIT 1;

Answer:

Country, Population
'China', '1277558000'
```

13. **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.

```
SELECT Name AS City, CountryCode
FROM City
WHERE CountryCode = 'ESP' AND Name = 'Madrid';

Answer:

City, CountryCode
'Madrid', 'ESP'
```

14. **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.

```
SELECT Name AS City, CountryCode, Population
FROM City
WHERE CountryCode IN ('AUT', 'BEL', 'BGR', 'HRV', 'CYP', 'CZE', 'DNK', 'EST', 'FIN', 'FRA',
'DEU', 'GRC', 'HUN', 'ISL', 'ITA', 'LVA', 'LTU', 'LUX', 'MLT', 'NLD', 'POL', 'PRT', 'ROU', 'SVK',
'SVN', 'ESP', 'SWE')
ORDER BY Name ASC;
```

Answer, 431 rows returned, these are some:

City, CountryCode, Population
'[San Cristóbal de] la Laguna', 'ESP', '127945'
'´s-Hertogenbosch', 'NLD', '129170'
'A Coruña (La Coruña)', 'ESP', '243402'
'Aachen', 'DEU', '243825'
'Aalborg', 'DNK', '161161'
'Aix-en-Provence', 'FRA', '134222'
'Albacete', 'ESP', '147527'
'Alcalá de Henares', 'ESP', '164463'
'Alcorcón', 'ESP', '142048'
'Alessandria', 'ITA', ‘90289'

15. **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.

```
SELECT CountryCode, AVG(Population) AS AveragePopulation
FROM City
GROUP BY CountryCode
ORDER BY AveragePopulation DESC;
```

Answer, 232 rows returned, these are some:

CountryCode, AveragePopulation
'SGP', '4017733.0000'
'HKG', '1650316.5000'
'URY', '1236000.0000'
'GIN', '1090610.0000'
'UGA', '890800.0000'
'LBR', '850000.0000'
'SLE', '850000.0000'
'MLI', '809552.0000'
'AUS', '808119.0000'
'MNG', '773700.0000'

16. **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.

```
SELECT Country.Name AS CountryName, City.Name AS CapitalCity, City.Population
AS CapitalPopulation
FROM Country
JOIN City ON City.ID = Country.Capital
ORDER BY City.Population DESC;

Answer, 232 rows returned, here are some;

CountryName, CapitalCity, CapitalPopulation
'South Korea', 'Seoul', '9981619'
'Indonesia', 'Jakarta', '9604900'
'Mexico', 'Ciudad de México', '8591309'
'Russian Federation', 'Moscow', '8389200'
'Japan', 'Tokyo', '7980230'
'China', 'Peking', '7472000'
'United Kingdom', 'London', '7285000'
'Egypt', 'Cairo', '6789479'
'Iran', 'Teheran', '6758845'
'Peru', 'Lima', '6464693'
```

17. **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.

```
SELECT Name AS CountryName, Population, SurfaceArea,
      (Population / SurfaceArea) AS PopulationDensity
FROM Country
WHERE SurfaceArea > 0
ORDER BY PopulationDensity ASC;
```

Answer, 239 returned, here are some:

```
CountryName, Population, SurfaceArea, PopulationDensity
'Antarctica', '0', '13120000.00', '0.0000'
'Bouvet Island', '0', '59.00', '0.0000'
'South Georgia and the South Sandwich Islands', '0', '3903.00', '0.0000'
'British Indian Ocean Territory', '0', '78.00', '0.0000'
'Heard Island and McDonald Islands', '0', '359.00', '0.0000'
'United States Minor Outlying Islands', '0', '16.00', '0.0000'
'French Southern territories', '0', '7780.00', '0.0000'
'Greenland', '56000', '2166090.00', '0.0259'
'Svalbard and Jan Mayen', '3200', '62422.00', '0.0513'
'Falkland Islands', '2000', '12173.00', '0.1643'
```

18. **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.

```
SELECT c.Name AS CityName, c.Population, co.GNP AS GDP,
      (co.GNP / c.Population) AS GDPPerCapita
FROM City c
JOIN Country co ON c.CountryCode = co.Code
WHERE c.Population > 0
HAVING (co.GNP / c.Population) > (
   SELECT AVG(co.GNP / c.Population)
   FROM City c
   JOIN Country co ON c.CountryCode = co.Code
   WHERE c.Population > 0
)
ORDER BY GDPPerCapita DESC;
```

Answer, 842 rows returned, here are some:

```
CityName, Population, GDP, GDPPerCapita
'Charleston', '89063', '8510700.00', '95.558200'
'Carson', '89089', '8510700.00', '95.530312'
'Odessa', '89293', '8510700.00', '95.312063'
'Elgin', '89408', '8510700.00', '95.189469'
'Kenosha', '89447', '8510700.00', '95.147965'
'Fall River', '90555', '8510700.00', '93.983767'
'Santa Monica', '91084', '8510700.00', '93.437925'
'Cary', '91213', '8510700.00', '93.305779'
'Boulder', '91238', '8510700.00', '93.280212'
'Visalia', '91762', '8510700.00', '92.747543'
```

19. **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.

```
SELECT Name AS City, CountryCode, Population
FROM City
ORDER BY Population DESC
LIMIT 10 OFFSET 30;
```

Answer:

```
 City, CountryCode, Population
'Shenyang', 'CHN', '4265200'
'Kanton [Guangzhou]', 'CHN', '4256300'
'Singapore', 'SGP', '4017733'
'Ho Chi Minh City', 'VNM', '3980000'
'Chennai (Madras)', 'IND', '3841396'
'Pusan', 'KOR', '3804522'
'Los Angeles', 'USA', '3694820'
'Dhaka', 'BGD', '3612850'
'Berlin', 'DEU', '3386667'
'Rangoon (Yangon)', 'MMR', '3361700'
```

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:

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.