In [1]:
```
#Q1
```

In [2]:
```
# A database is an organized collection of structured information, or data, typical

# Differentiate between sql n no sql:=

# SQL:-These databases have fixed or static or predefined schema.These databases ar
# Examples: MySQL, PostgreSQL, Oracle, MS-SQL Server, etc


#NoSql:- Non-relational or distributed database system.Non-relational or distribute
# Horizontally scalable.Follows CAP(consistency, availability, partition tolerance)
# Examples: MongoDB, GraphQL, HBase, Neo4j, Cassandra, etc
```

In [3]:
```
#Q2
```

In [11]:
```
# DDL:- DDL dealing with database Schemas, as well as the description of how data r
# Ex:- Create,Alter,Drop,rename,Truncate and Comment.

#The CREATE TABLE command creates a new table in the database.
#Ex:-
#CREATE TABLE Persons (
  #  PersonID int,
    #LastName varchar(255),
    #FirstName varchar(255),
   # Address varchar(255),
  #  City varchar(255)

 #  )


#The ALTER TABLE command adds, deletes, or modifies columns in a table.
#ALTER TABLE Customers
#ADD Email varchar(255);


#The DROP TABLE command deletes a table in the database.
# DROP TABLE Shippers;


#The TRUNCATE TABLE command deletes the data inside a table, but not the table itse
# TRUNCATE TABLE Categories;
```

In [12]:
```
#Q3
```

In [13]:
```
# DML denotes data manipulation language which includes commands such as select,
# Update,delete,merge,cell,locktable.


#The DELETE statement removes a row or combination of rows from a table.
# Ex
#DELETE tableName
#WHERE  filterColumn=filterValue;
```

```
#The UPDATE statement enables users to update a row or group of rows in a table.
# UPDATE dbo.Department
 #  SET GroupName = 'Room'
#WHERE GroupName = 'Ward'


#INSERT is for adding single or multiple rows to a table. INSERT can also help with
# ex
#INSERT INTO tableName
#(column1, column2, ...)
#VALUES (value1, value2, ...)
```

In [14]:
```
#Q4
```

In [15]:
```
#Data Query Language (DQL) is one of the basic sub-languages of SQL statements. The
#It is also occasionally suggested that a transaction control language (TCL) belong
#DQL statements are employed to conduct inquiries on the information contained in s


#The most utilized SQL statement is select. Data from a database can be retrieved o
# In the SELECT command, either we can show all the columns, or we can get some spe
#This analysis is very much needed because we have to be very cautious about which


#Ex:-
#Select * FROM tablename;
#SELECT * FROM columnname Where name = 'Char';
```

In [16]:
```
#Q5
```

In [17]:
```
#Primary Key:-It helps in the unique identifcation of data in a database.There can
# Null values are not acceptable.Primary key index is automaticallly created.


#Foreign Key:-It helps established a link between tables.There can be more than one
# is not created automatically.
```

In [20]:
```
#Q6
```

In [23]:
```
#import mysql.connector
# import mysql.connector
#create user 'user'@'%' identified by 'password'
#mydb = mysql.connector.connect(
#  host="localhost",
#  user="abc",
#  password="password"
#)
#print(mydb)
#mycursor = mydb.cursor()
#mycursor.execute("SHOW DATABASES")
#for x in mycursor:
#  print(x)
```

[Running] python -u "/config/workspace/demo.py" <mysql.connector.connection_cext.CMySQLConnection object at 0x7fbbc5649730> ('information_schema',) ('mysql',) ('performance_schema',) ('sys',) [Done] exited with code=0 in 0.116 seconds

In [24]:
```
#Q7
```

In [25]:
```
#SQL queries adhere to a specific order when evaluating clauses, similar to how mat

#From the eyes of the user, queries begin from the first clause and end at the last

#The order in which the clauses in queries are executed is as follows.



#1. FROM/JOIN
#The FROM and/or JOIN clauses are executed first to determine the data of interest.



#2. WHERE
#The WHERE clause is executed to filter out records that do not meet the constraint



#3. GROUP BY
#The GROUP BY clause is executed to group the data based on the values in one or mo



#4. HAVING
#The HAVING clause is executed to remove the created grouped records that don’t mee



#5. SELECT
#The SELECT clause is executed to derive all desired columns and expressions.



#6. ORDER BY
#The ORDER BY clause is executed to sort the derived values in ascending or descend

#7. LIMIT/OFFSET
#Finally, the LIMIT and/or OFFSET clauses are executed to keep or skip a specified
```

In [ ]:
```

```