**1.Write a Python program to create a MySQL database and a table.**

**solution:-**

import mysql.connector

mydb = mysql.connector.connect(

host="localhost",

user="yourusername",

password="yourpassword"

)

mycursor = mydb.cursor()

mycursor.execute("CREATE DATABASE mydatabase")

mycursor.execute("CREATE TABLE customers (name VARCHAR(255), address VARCHAR(255))")

**2. Write a Python program to insert data into a MySQL table.**

**solution:-**

import mysql.connector

mydb = mysql.connector.connect(

host="localhost",

user="yourusername",

password="yourpassword",

database="mydatabase"

)

mycursor = mydb.cursor()

sql = "INSERT INTO customers (name, address) VALUES (%s, %s)"

val = ("John", "Highway 21")

mycursor.execute(sql, val)

mydb.commit()

print(mycursor.rowcount, "record inserted.")

**3. Write a Python program to create an index on a MySQL table.**

**solution:-**

import mysql.connector

mydb = mysql.connector.connect(

host="localhost",

user="yourusername",

password="yourpassword",

database="mydatabase"

)

mycursor = mydb.cursor()

mycursor.execute("CREATE INDEX idx\_name ON customers (name)")

**4. Write a Python program to join two tables in MySQL.**

**solution:-**

import mysql.connector

mydb = mysql.connector.connect(

host="localhost",

user="yourusername",

password="yourpassword",

database="mydatabase"

)

mycursor = mydb.cursor()

mycursor.execute("SELECT customers.name, orders.product\_name FROM customers INNER JOIN orders ON customers.id = orders.customer\_id")

myresult = mycursor.fetchall()

for x in myresult:

print(x)

**5. Write a Python program to handle MySQL errors using exception handling.**

**solution:-**

import mysql.connector

try:

mydb = mysql.connector.connect(

host="localhost",

user="yourusername",

password="yourpassword",

database="mydatabase"

)

except mysql.connector.Error as err:

print("Something went wrong: {}".format(err))

else:

mycursor = mydb.cursor()

mycursor.execute("SELECT \* FROM customers")

myresult = mycursor.fetchall()

for x in myresult:

print(x)

finally:

mydb.close()

**6. Write a Python program to connect to a MongoDB database and insert data.**

**solution:-**

import pymongo

myclient = pymongo.MongoClient("mongodb://localhost:27017/")

mydb = myclient["mydatabase"]

mycol = mydb["customers"]

mydict = { "name": "John", "address": "Highway 37" }

x = mycol.insert\_one(mydict)

print(x.inserted\_id)

**7. Write a Python program to update data in a MongoDB database.**

**solution:-**

import pymongo

myclient = pymongo.MongoClient("mongodb://localhost:27017/")

mydb = myclient["mydatabase"]

mycol = mydb["customers"]

myquery = { "address": "Highway 37" }

newvalues = { "$set": { "address": "Park Lane 38" } }

mycol.update\_many(myquery, newvalues)

print("Documents updated:", mycol.modified\_count)

**8. Write a Python program to handle MongoDB errors using exception handling.**

**solution:-**

import pymongo

try:

myclient = pymongo.MongoClient("mongodb://localhost:27017/")

mydb = myclient["mydatabase"]

mycol = mydb["customers"]

except pymongo.errors.ConnectionFailure as err:

print("Could not connect to MongoDB: {}".format(err))

else:

mydict = { "name": "John", "address": "Highway 37" }

x = mycol.insert\_one(mydict)

print(x.inserted\_id)

finally:

myclient.close()

**9. Write a Python program to query a MongoDB database using aggregation.**

**solution:-**

import pymongo

myclient = pymongo.MongoClient("mongodb://localhost:27017/")

mydb = myclient["mydatabase"]

mycol = mydb["customers"]

pipeline = [

{ "$match": { "address": "Highway 37" } },

{ "$group": { "\_id": "$name", "count": { "$sum": 1 } } }

]

result = mycol.aggregate(pipeline)

for x in result:

print(x)

**10. Write a Python program to perform a text search on a MongoDB database.**

**solution:-**

import pymongo

myclient = pymongo.MongoClient("mongodb://localhost:27017/")

mydb = myclient["mydatabase"]

mycol = mydb["customers"]

myquery = { "$text": { "$search": "apple" } }

mydoc = mycol.find(myquery)

for x in mydoc:

print(x)