

Name: Rohit Mukund Modak

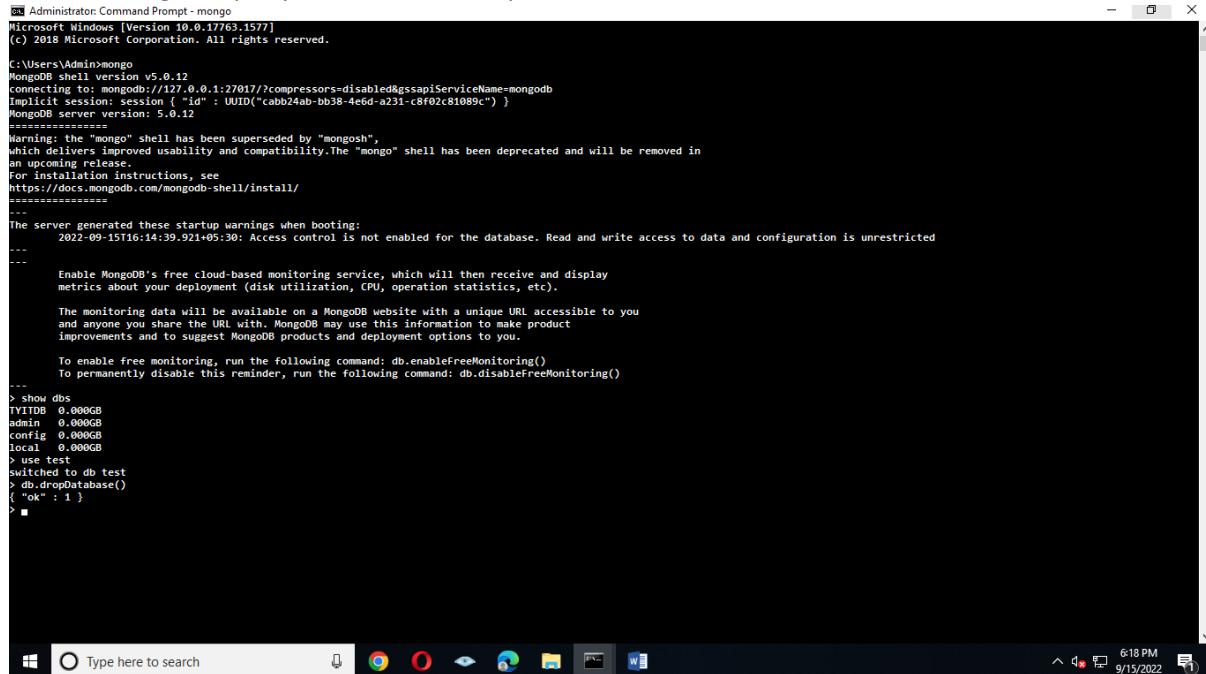
Roll No.: 229667

Subject: Next Generation Technologies Practical

Index			
Sr No.	Date	Topic	Page No.
1		MongoDB Basics	
2		Simple Queries with MongoDB	
3		Implementing Aggregation	
4		Replication, Backup and Restore	
5		Java and MongoDB	
6		PHP and MongoDB	
7		Python and MongoDB	
8		Programs on Basic jQuery	
9		jQuery Advanced	
10		JSON	
11		Create a JSON file and import it to MongoDB	

Sr No.	Date	Topic	Page No.
1		MongoDB Basics	

1. Write a mongodb query to create and drop database:



```

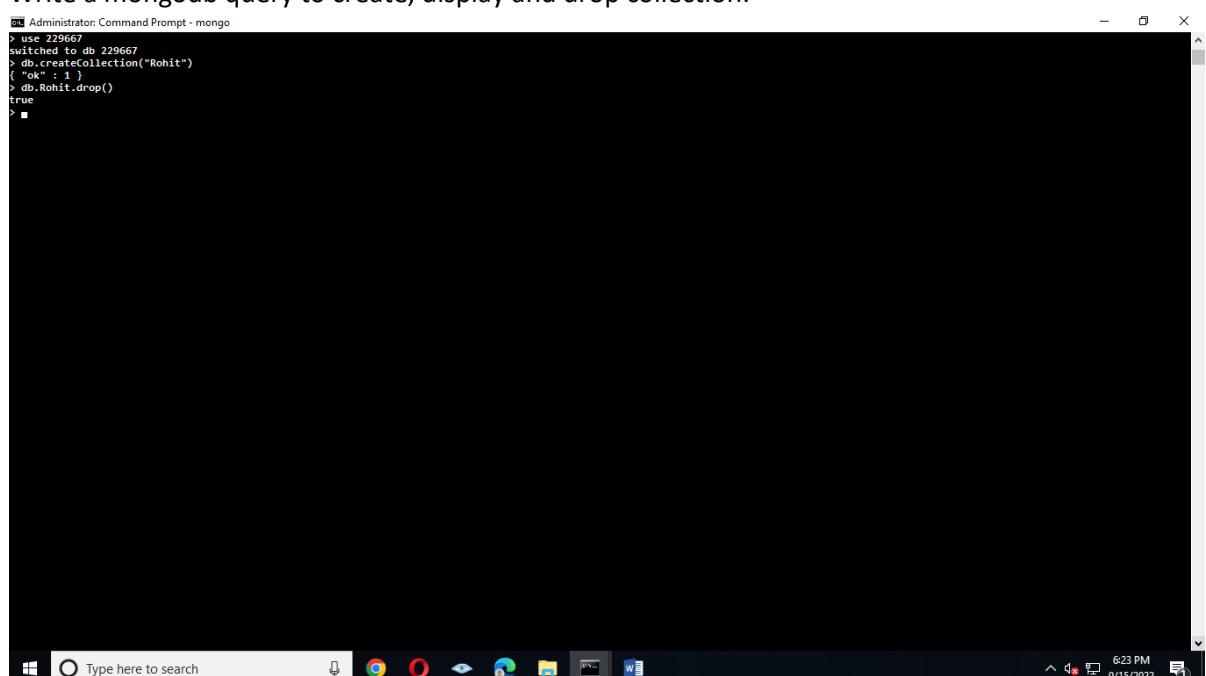
Administrator: Command Prompt - mongo
Microsoft Windows [Version 10.0.17763.1577]
(c) 2018 Microsoft Corporation. All rights reserved.

C:\Users\Admin>mongo
MongoDB shell version v5.0.12
connecting to: mongodb://127.0.0.1:27017/?compressors=disabled&gssapiServiceName=mongodb
Implicit session: session { "id" : UUID("cab024ab-bb38-4e6d-a231-c8f02c81089c") }
MongoDB server version: 5.0.12
-----
Warning: the "mongo" shell has been superseded by "mongosh",
which delivers improved usability and compatibility. The "mongo" shell has been deprecated and will be removed in
an upcoming release.
For installation instructions, see
https://docs.mongodb.com/mongodb-shell/install/
-----
The server generated these startup warnings when booting:
2022-09-15T16:14:39.921+05:30: Access control is not enabled for the database. Read and write access to data and configuration is unrestricted
...
...
Enable MongoDB's free cloud-based monitoring service, which will then receive and display
metrics about your deployment (disk utilization, CPU, operation statistics, etc).
The monitoring data will be available on a MongoDB website with a unique URL accessible to you
and anyone you share the URL with. MongoDB may use this information to make product
improvements and to suggest MongoDB products and deployment options to you.

To enable free monitoring, run the following command: db.enableFreeMonitoring()
To permanently disable this reminder, run the following command: db.disableFreeMonitoring()
...
> show dbs
test 0.000GB
local 0.000GB
config 0.000GB
local 0.000GB
> use test
switched to db test
> db.dropDatabase()
{ "ok" : 1 }
> 

```

2. Write a mongodb query to create, display and drop collection:



```

Administrator: Command Prompt - mongo
> use 229667
switched to db 229667
> db.createCollection("Rohit")
{ "ok" : 1 }
> db.Rohit.drop()
true
> 

```

3. Write a mongodb query to insert, query, update and delete a document:

```
Administrator: Command Prompt - mongo
> <*> createCollection("Rohit")
> <*>
> db.Rohit.insert({COURSE:"TYIT",COLLEGE:"MCC"})
WriteResult({ "nInserted" : 1 })
> db.Rohit.find().pretty()
{
    "_id" : ObjectId("632320bcbbb0aa0bd626a315"),
    "COURSE" : "TYIT",
    "COLLEGE" : "MCC"
}
> db.Rohit.update({},{$set:{"COLLEGE":"MULUND COLLEGE OF COMMERCE"}})
WriteResult({ "nMatched" : 1, "nUpserted" : 0, "nModified" : 1 })
> db.Rohit.find().pretty()
{
    "_id" : ObjectId("632320bcbbb0aa0bd626a315"),
    "COURSE" : "TYIT",
    "COLLEGE" : "MULUND COLLEGE OF COMMERCE"
}
> db.Rohit.remove({})
WriteResult({ "nRemoved" : 1 })
> db.Rohit.find().pretty()
> ■
```

Sr No.	Date	Topic	Page No.
2		Simple Queries with MongoDB	

1. Insert the documents:

```
Administrator: Command Prompt - mongo
local 0.000GB
> use TYITDB
switched to db TYITDB
> db.createCollection(TYITCOL)
uncaught exception: ReferenceError: TYITCOL is not defined :
@(shell):1:1
> db.createCollection('TYITCOL')
{
  "ok" : 1
}
> db.TYITCOL.insertMany([{"FNAME":"ROHIT",LNAME:"MODAK",AGE:20,GENDER:"M",COUNTRY:"INDIA"}, {"FNAME":"RAHUL",LNAME:"KADAM",AGE:20,GENDER:"M",COUNTRY:"UK"} , {"FNAME":"AARTI",LNAME:"CHAUDHARI",AGE:19,GENDER:"F",COUNTRY:"USA"}, {"FNAME":"LEENA",LNAME:"CHAUDHARI",AGE:18,GENDER:"F",COUNTRY:"INDIA"}, {"FNAME":"SANDHYA",LNAME:"CHAUDHARI",AGE:35,GENDER:"F",COUNTRY:"UK"}])
{
  "acknowledged" : true,
  "insertedIds" : [
    ObjectId("63230632b6f8a40664a0b6f4"),
    ObjectId("63230632b6f8a40664a0b6f5"),
    ObjectId("63230632b6f8a40664a0b6f6"),
    ObjectId("63230632b6f8a40664a0b6f7"),
    ObjectId("63230632b6f8a40664a0b6f8")
  ]
}
> db.TYITCOL.insertMany([{"FNAME":"AAROHI",LNAME:"MODAK",AGE:20,GENDER:"F",COUNTRY:"INDIA"}, {"FNAME":"TANVI",LNAME:"KADAM",AGE:25,GENDER:"F",COUNTRY:"USA"} , {"FNAME":"ARCHANA",LNAME:"MODAK",AGE:45,GENDER:"F",COUNTRY:"INDIA"}, {"FNAME":"MUKUND",LNAME:"MODAK",AGE:55,GENDER:"M",COUNTRY:"UK"}, {"FNAME":"CHARUDATT",LNAME:"CHAUDHARI",AGE:44,GENDER:"M",COUNTRY:"USA"}])
{
  "acknowledged" : true,
  "insertedIds" : [
    ObjectId("6323071bb6f8a40664a0b6f9"),
    ObjectId("6323071bb6f8a40664a0b6fa"),
    ObjectId("6323071bb6f8a40664a0b6fb"),
    ObjectId("6323071bb6f8a40664a0b6fc"),
    ObjectId("6323071bb6f8a40664a0b6fd")
  ]
}
> ■
Windows Type here to search 4:37 PM 9/15/2022
```

```
Administrator: Command Prompt - mongo
> db.TYITCOL.find().pretty()
{
  "_id" : ObjectId("63230632b6f8a40664a0b6f4"),
  "FNAME" : "ROHIT",
  "LNAME" : "MODAK",
  "AGE" : 20,
  "GENDER" : "M",
  "COUNTRY" : "INDIA"
}

{
  "_id" : ObjectId("63230632b6f8a40664a0b6f5"),
  "FNAME" : "RAHUL",
  "LNAME" : "KADAM",
  "AGE" : 20,
  "GENDER" : "M",
  "COUNTRY" : "UK"
}

{
  "_id" : ObjectId("63230632b6f8a40664a0b6f6"),
  "FNAME" : "AARTI",
  "LNAME" : "CHAUDHARI",
  "AGE" : 19,
  "GENDER" : "F",
  "COUNTRY" : "USA"
}

{
  "_id" : ObjectId("63230632b6f8a40664a0b6f7"),
  "FNAME" : "LEENA",
  "LNAME" : "CHAUDHARI",
  "AGE" : 18,
  "GENDER" : "F",
  "COUNTRY" : "INDIA"
}

{
  "_id" : ObjectId("63230632b6f8a40664a0b6f8"),
  "FNAME" : "SANDHYA",
  "LNAME" : "CHAUDHARI",
  "AGE" : 35,
  "GENDER" : "F",
  "COUNTRY" : "UK"
}

{
  "_id" : ObjectId("6323071bb6f8a40664a0b6f9"),
  "FNAME" : "AAROHI",
  "LNAME" : "MODAK",
  "AGE" : 20,
  "GENDER" : "F",
  "COUNTRY" : "INDIA"
}

{
  "_id" : ObjectId("6323071bb6f8a40664a0b6fa"),
  "FNAME" : "TANVI",
  "LNAME" : "KADAM",
  "AGE" : 25,
  "GENDER" : "F",
  "COUNTRY" : "INDIA"
}

{
  "_id" : ObjectId("6323071bb6f8a40664a0b6fb"),
  "FNAME" : "ARCHANA",
  "LNAME" : "MODAK",
  "AGE" : 45,
  "GENDER" : "F",
  "COUNTRY" : "INDIA"
}

{
  "_id" : ObjectId("6323071bb6f8a40664a0b6fc"),
  "FNAME" : "MUKUND",
  "LNAME" : "MODAK",
  "AGE" : 55,
  "GENDER" : "M",
  "COUNTRY" : "UK"
}

{
  "_id" : ObjectId("6323071bb6f8a40664a0b6fd"),
  "FNAME" : "CHARUDATT",
  "LNAME" : "CHAUDHARI",
  "AGE" : 44,
  "GENDER" : "M",
  "COUNTRY" : "USA"
}
Windows Type here to search 4:52 PM 9/15/2022
```

2. Update the country to UK for all female users:

```
Administrator: Command Prompt - mongo
> db.TYITCOL.updateMany({GENDER:"F"},{$set:{COUNTRY:"UK"}})
{ "acknowledged" : true, "matchedCount" : 6, "modifiedCount" : 6 }
> db.TYITCOL.find(GENDER:"F").pretty()
{
  "_id" : ObjectId("63230632b6f8a40664a0b6f6"),
  "FNAME" : "ARTI",
  "LNAME" : "CHAUDHARI",
  "AGE" : 19,
  "GENDER" : "F",
  "COUNTRY" : "UK"
}
{
  "_id" : ObjectId("63230632b6f8a40664a0b6f7"),
  "FNAME" : "LEENA",
  "LNAME" : "CHAUDHARI",
  "AGE" : 18,
  "GENDER" : "F",
  "COUNTRY" : "UK"
}
{
  "_id" : ObjectId("63230632b6f8a40664a0b6f8"),
  "FNAME" : "SANDHYA",
  "LNAME" : "CHAUDHARI",
  "AGE" : 35,
  "GENDER" : "F",
  "COUNTRY" : "UK"
}
{
  "_id" : ObjectId("6323071bb6f8a40664a0b6f9"),
  "FNAME" : "AAROHI",
  "LNAME" : "MODAK",
  "AGE" : 20,
  "GENDER" : "F",
  "COUNTRY" : "UK"
}
{
  "_id" : ObjectId("6323071bb6f8a40664a0b6fa"),
  "FNAME" : "TANVI",
  "LNAME" : "KADAM",
  "AGE" : 25,
  "GENDER" : "F",
  "COUNTRY" : "UK"
}
{
  "_id" : ObjectId("6323071bb6f8a40664a0b6fb"),
  "FNAME" : "ARCHANA",
  "LNAME" : "MODAK",
  "AGE" : 45,
  "GENDER" : "F",
  "COUNTRY" : "UK"
}
Windows Type here to search 4:50 PM 9/15/2022
```

3. Add the new field company to all the documents:

```
Administrator: Command Prompt - mongo
> db.TYITCOL.updateMany({},{$set:{COMPANY:"TECHMAX"}})
{ "acknowledged" : true, "matchedCount" : 10, "modifiedCount" : 10 }
> db.TYITCOL.find().pretty()
{
  "_id" : ObjectId("63230632b6f8a40664a0b6f4"),
  "FNAME" : "ROHIT",
  "LNAME" : "MODAK",
  "AGE" : 20,
  "GENDER" : "M",
  "COUNTRY" : "INDIA",
  "COMPANY" : "TECHMAX"
}
{
  "_id" : ObjectId("63230632b6f8a40664a0b6f5"),
  "FNAME" : "RAHUL",
  "LNAME" : "KADAM",
  "AGE" : 20,
  "GENDER" : "M",
  "COUNTRY" : "UK",
  "COMPANY" : "TECHMAX"
}
{
  "_id" : ObjectId("63230632b6f8a40664a0b6f6"),
  "FNAME" : "ARTI",
  "LNAME" : "CHAUDHARI",
  "AGE" : 19,
  "GENDER" : "F",
  "COUNTRY" : "UK",
  "COMPANY" : "TECHMAX"
}
{
  "_id" : ObjectId("63230632b6f8a40664a0b6f7"),
  "FNAME" : "LEENA",
  "LNAME" : "CHAUDHARI",
  "AGE" : 18,
  "GENDER" : "F",
  "COUNTRY" : "UK",
  "COMPANY" : "TECHMAX"
}
{
  "_id" : ObjectId("63230632b6f8a40664a0b6f8"),
  "FNAME" : "SANDHYA",
  "LNAME" : "CHAUDHARI",
  "AGE" : 35,
  "GENDER" : "F",
  "COUNTRY" : "UK",
  "COMPANY" : "TECHMAX"
}
{
  "_id" : ObjectId("6323071bb6f8a40664a0b6f9"),
  Windows Type here to search 4:55 PM 9/15/2022
```

4. Delete all the documents where gender = "m"

```
Administrator: Command Prompt - mongo
> db.TYITCOL.remove({GENDER: "M"})
WriteResult({ "nRemoved" : 4 })
> db.TYITCOL.find().pretty()
{
    "_id" : ObjectId("63230632b6f8a40664a0b6f6"),
    "FNAME" : "ARTI",
    "LNAME" : "CHAUDHARI",
    "AGE" : 19,
    "GENDER" : "F",
    "COUNTRY" : "USA",
    "COMPANY" : "TECHMAX"
}
{
    "_id" : ObjectId("63230632b6f8a40664a0b6f7"),
    "FNAME" : "LEENA",
    "LNAME" : "CHAUDHARI",
    "AGE" : 18,
    "GENDER" : "F",
    "COUNTRY" : "USA",
    "COMPANY" : "TECHMAX"
}
{
    "_id" : ObjectId("63230632b6f8a40664a0b6f8"),
    "FNAME" : "SANDHYA",
    "LNAME" : "CHAUDHARI",
    "AGE" : 35,
    "GENDER" : "F",
    "COUNTRY" : "USA",
    "COMPANY" : "TECHMAX"
}
{
    "_id" : ObjectId("6323071bb6f8a40664a0b6f9"),
    "FNAME" : "AAROHI",
    "LNAME" : "MODAK",
    "AGE" : 20,
    "GENDER" : "F",
    "COUNTRY" : "USA",
    "COMPANY" : "TECHMAX"
}
{
    "_id" : ObjectId("6323071bb6f8a40664a0b6fa"),
    "FNAME" : "TANVI",
    "LNAME" : "KADAM",
    "AGE" : 25,
    "GENDER" : "F",
    "COUNTRY" : "USA",
    "COMPANY" : "TECHMAX"
}
{
    "_id" : ObjectId("6323071bb6f8a40664a0b6fb"),
    "FNAME" : "SHREYA",
    "LNAME" : "KADAM",
    "AGE" : 28,
    "GENDER" : "F",
    "COUNTRY" : "INDIA",
    "COMPANY" : "TECHMAX"
}
```

5. Find out a count of female users who stay in either India or USA:

```
Administrator: Command Prompt - mongo
> db.TYITCOL.find({GENDER: "F", $or: [{COUNTRY: "USA"}, {COUNTRY: "INDIA"}]}).count()
6
```

6. Display the first name and age of all female employees

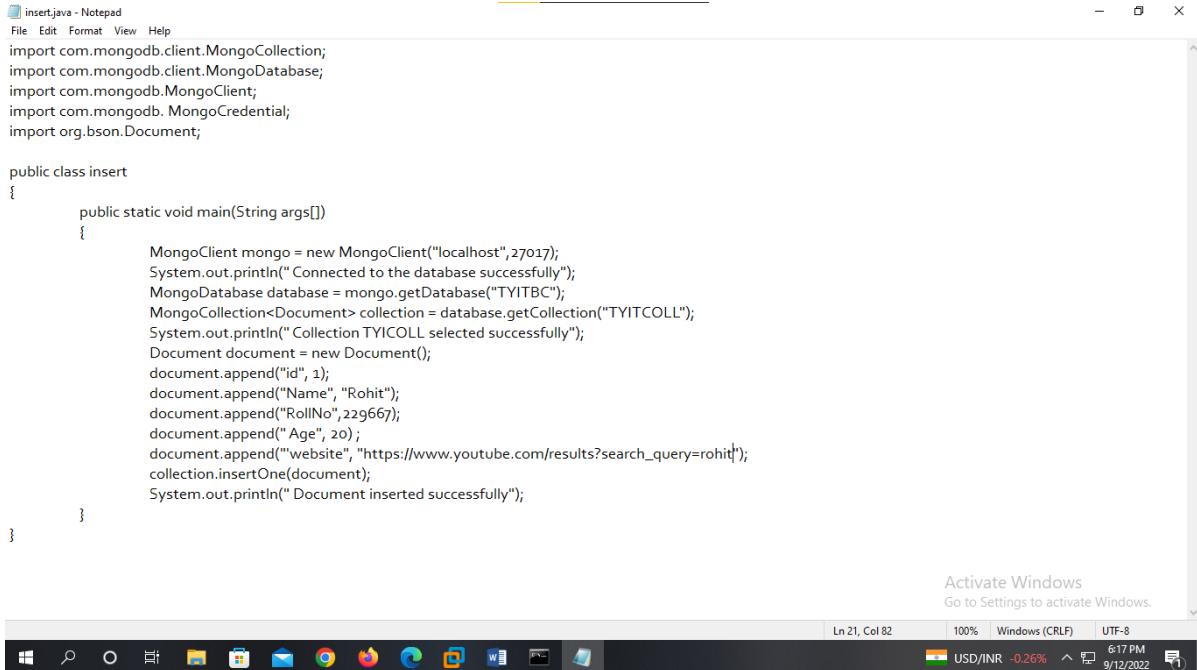
```
Administrator: Command Prompt - mongo
> db.TYITCOL.find({GENDER:"F"},{FNAME:1,AGE:1})
{
  "_id": ObjectId("63230632b6f8a40664a0b6fb"),
  "NAME": "ARTI",
  "AGE": 19
},
{
  "_id": ObjectId("63230632b6f8a40664a0b6ff"),
  "NAME": "LEENA",
  "AGE": 18
},
{
  "_id": ObjectId("6323071bb6f8a40664a0b6f9"),
  "NAME": "SHREYA",
  "AGE": 25
},
{
  "_id": ObjectId("6323071bb6f8a40664a0b6fa"),
  "NAME": "AAROHI",
  "AGE": 20
},
{
  "_id": ObjectId("6323071bb6f8a40664a0b6fb"),
  "NAME": "TANVI",
  "AGE": 25
},
{
  "_id": ObjectId("6323071bb6f8a40664a0b6fc"),
  "NAME": "ARCHANA",
  "AGE": 45
}
>
```

Sr No.	Date	Topic	Page No.
3		Implementing Aggregation	

Sr No.	Date	Topic	Page No.
4		Replication, Backup and Restore	

Sr No.	Date	Topic	Page No.
5		Java and MongoDB	

1. Inserting the document:



```

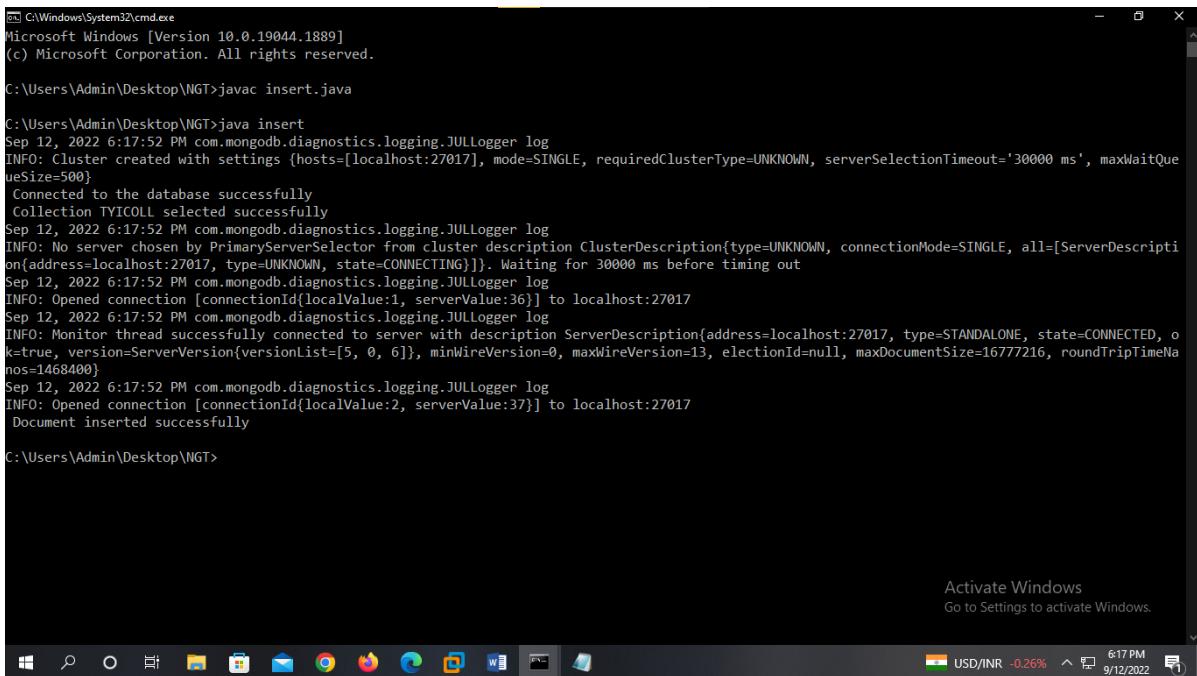
insert.java - Notepad
File Edit Format View Help
import com.mongodb.client.MongoCollection;
import com.mongodb.client.MongoDatabase;
import com.mongodb.MongoClient;
import com.mongodb.MongoCredential;
import org.bson.Document;

public class insert
{
    public static void main(String args[])
    {
        MongoClient mongo = new MongoClient("localhost",27017);
        System.out.println("Connected to the database successfully");
        MongoDatabase database = mongo.getDatabase("TYITBC");
        MongoCollection<Document> collection = database.getCollection("TYICOLL");
        System.out.println("Collection TYICOLL selected successfully");
        Document document = new Document();
        document.append("id", 1);
        document.append("Name", "Rohit");
        document.append("RollNo", 229667);
        document.append("Age", 20);
        document.append("website", "https://www.youtube.com/results?search_query=rohit");
        collection.insertOne(document);
        System.out.println(" Document inserted successfully");
    }
}

```

Activate Windows
Go to Settings to activate Windows.

Ln 21, Col 82 | 100% | Windows (CRLF) | UTF-8
USD/INR -0.26% 6:17 PM 9/12/2022



```

C:\Windows\System32\cmd.exe
Microsoft Windows [Version 10.0.19044.1889]
(c) Microsoft Corporation. All rights reserved.

C:\Users\Admin\Desktop\NGT>javac insert.java

C:\Users\Admin\Desktop\NGT>java insert
Sep 12, 2022 6:17:52 PM com.mongodb.diagnostics.logging.JULLoader log
INFO: Cluster created with settings {hosts=[localhost:27017], mode=SINGLE, requiredClusterType=UNKNOWN, serverSelectionTimeout='30000 ms', maxWaitQueueSize=500}
Connected to the database successfully
Collection TYICOLL selected successfully
Sep 12, 2022 6:17:52 PM com.mongodb.diagnostics.logging.JULLoader log
INFO: No server chosen by PrimaryServerSelector from cluster description ClusterDescription{type=UNKNOWN, connectionMode=SINGLE, all=[ServerDescription{address=localhost:27017, type=UNKNOWN, state=CONNECTING}]. Waiting for 30000 ms before timing out
Sep 12, 2022 6:17:52 PM com.mongodb.diagnostics.logging.JULLoader log
INFO: Opened connection [connectionId{localValue:1, serverValue:36}] to localhost:27017
Sep 12, 2022 6:17:52 PM com.mongodb.diagnostics.logging.JULLoader log
INFO: Monitor thread successfully connected to server with description ServerDescription{address=localhost:27017, type=STANDALONE, state=CONNECTED, ok=true, version=ServerVersion{versionList=[5, 0, 6]}, minWireVersion=0, maxWireVersion=13, electionId=null, maxDocumentSize=16777216, roundTripTimeNanos=1468400}
Sep 12, 2022 6:17:52 PM com.mongodb.diagnostics.logging.JULLoader log
INFO: Opened connection [connectionId{localValue:2, serverValue:37}] to localhost:27017
Document inserted successfully

C:\Users\Admin\Desktop\NGT>

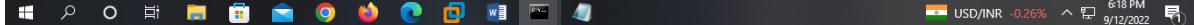
```

Activate Windows
Go to Settings to activate Windows.

Ln 21, Col 82 | 100% | Windows (CRLF) | UTF-8
USD/INR -0.26% 6:17 PM 9/12/2022

```
 Command Prompt - mongo
> db.TYITCOLL.find().pretty()
> db.TYITCOLL.find().pretty()
{
    "_id" : ObjectId("631f2a78ff7edalc897a42bd"),
    "id" : 1,
    "Name" : "Rohit",
    "RollNo" : 229667,
    "Age" : 20,
    "website" : "https://www.youtube.com/results?search_query=rohit"
}
>
```

Activate Windows
Go to Settings to activate Windows.



2. Updating the document:

```
updatejava - Notepad
File Edit Format View Help
import com.mongodb.client.MongoCollection;
import com.mongodb.BasicDBObject;
import com.mongodb.DB;
import com.mongodb.DBCollection;
import com.mongodbDBObject;
import com.mongodb.MongoClient;
import com.mongodb.WriteResult;
import java.net.UnknownHostException;

public class update
{
    public static void main(String args[])
    {
        MongoClient mongo = new MongoClient("localhost", 27017);
        System.out.println("Connected to the database successfully");
        DB db = mongo.getDB("TYITBC");
        DBCollection col = db.getCollection("TYITCOLL");
       DBObject query = new BasicDBObject("id", 1);
       DBObject update = new BasicDBObject();
        update.put("$set", new BasicDBObject("Age", 35));
        WriteResult result = col.update(query, update);
        mongo.close();
    }
}
```

Activate Windows
Go to Settings to activate Windows.



```
C:\Windows\System32\cmd.exe
Microsoft Windows [Version 10.0.19044.1889]
(c) Microsoft Corporation. All rights reserved.

C:\Users\Admin\Desktop\NGT>javac update.java
Note: update.java uses or overrides a deprecated API.
Note: Recompile with -Xlint:deprecation for details.

C:\Users\Admin\Desktop\NGT>java update
Sep 12, 2022 6:18:56 PM com.mongodb.diagnostics.logging.JULLogger log
INFO: Cluster created with settings {hosts=[localhost:27017], mode=SINGLE, requiredClusterType=UNKNOWN, serverSelectionTimeout='30000 ms', maxWaitQueueSize=500}
Connected to the database successfully
Sep 12, 2022 6:18:56 PM com.mongodb.diagnostics.logging.JULLogger log
INFO: No server chosen by PrimaryServerSelector from cluster description ClusterDescription{type=UNKNOWN, connectionMode=SINGLE, all=[ServerDescription{address=localhost:27017, type=UNKNOWN, state=CONNECTING}]. Waiting for 30000 ms before timing out
Sep 12, 2022 6:18:56 PM com.mongodb.diagnostics.logging.JULLogger log
INFO: Opened connection [connectionId{localValue:1, serverValue:38}] to localhost:27017
Sep 12, 2022 6:18:56 PM com.mongodb.diagnostics.logging.JULLogger log
INFO: Monitor thread successfully connected to server with description ServerDescription{address=localhost:27017, type=STANDALONE, state=CONNECTED, ok=true, version=ServerVersion{versionList=[5, 0, 6]}, minWireVersion=0, maxWireVersion=13, electionId=null, maxDocumentSize=16777216, roundTripTimeNanos=600000}
Sep 12, 2022 6:18:56 PM com.mongodb.diagnostics.logging.JULLogger log
INFO: Opened connection [connectionId{localValue:2, serverValue:39}] to localhost:27017
Sep 12, 2022 6:18:56 PM com.mongodb.diagnostics.logging.JULLogger log
INFO: Closed connection [connectionId{localValue:2, serverValue:39}] to localhost:27017 because the pool has been closed.

C:\Users\Admin\Desktop\NGT>
```

Activate Windows
Go to Settings to activate Windows.

```
C:\Windows\System32\cmd.exe
MongoDB shell version v4.4.1
connecting to: mongodb://127.0.0.1:27017/
Implicit session: ticket does not exist.
MongoDB shell version v4.4.1
connecting to: mongodb://127.0.0.1:27017/
Implicit session: ticket does not exist.
> db.TVITCOLL.find().pretty()
{
    "_id" : ObjectId("631f2a78ff7eda1c807a42bd"),
    "id" : 1,
    "Name" : "Rohit",
    "RollNo" : 229667,
    "Age" : 35,
    "website" : "https://www.youtube.com/results?search_query=rohit"
}
>
```

Activate Windows
Go to Settings to activate Windows.

3. Retrieving the information from the document:

```

retrieve.java - Notepad
File Edit Format View Help
import com.mongodb.client.MongoCollection;
import com.mongodb.client.MongoDatabase;
import com.mongodb.MongoClient;
import com.mongodb.MongoCredential;
import org.bson.Document;
import java.util.Iterator;
import com.mongodb.client.FindIterable;

public class retrieve
{
    public static void main(String args[])
    {
        MongoClient mongo = new MongoClient("localhost",27017);
        System.out.println("Connected to the database successfully");
        MongoDatabase database = mongo.getDatabase("TYITBC");
        MongoCollection<Document> collection = database.getCollection("TYICOLL");
        System.out.println("Collection TYICOLL selected successfully");
        FindIterable<Document> iterDoc = collection.find();
        int i=1;
        Iterator it = iterDoc.iterator();
        while(it.hasNext())
        {
            System.out.println(it.next());
            i++;
        }
    }
}

```

Activate Windows
Go to Settings to activate Windows.

Ln 1, Col 1 | 100% | Windows (CRLF) | UTF-8
28°C Rain 6:19 PM 9/12/2022

```

C:\Windows\System32\cmd.exe
Microsoft Windows [Version 10.0.19044.1889]
(c) Microsoft Corporation. All rights reserved.

C:\Users\Admin\Desktop\NGT>javac retrieve.java

C:\Users\Admin\Desktop\NGT>java retrieve
Sep 12, 2022 6:19:55 PM com.mongodb.diagnostics.logging.JULLoader log
INFO: Cluster created with settings {hosts=[localhost:27017], mode=SINGLE, requiredClusterType=UNKNOWN, serverSelectionTimeout='30000 ms', maxWaitQueueSize=500}
Connected to the database successfully
Collection TYICOLL selected successfully
Sep 12, 2022 6:19:56 PM com.mongodb.diagnostics.logging.JULLoader log
INFO: No server chosen by ReadPreferenceServerSelector{readPreference=primary} from cluster description ClusterDescription{type=UNKNOWN, connectionMode=SINGLE, all=[ServerDescription{address=localhost:27017, type=UNKNOWN, state=CONNECTING}]. Waiting for 30000 ms before timing out
Sep 12, 2022 6:19:56 PM com.mongodb.diagnostics.logging.JULLoader log
INFO: Opened connection [connectionId{localValue:1, serverValue:40}] to localhost:27017
Sep 12, 2022 6:19:56 PM com.mongodb.diagnostics.logging.JULLoader log
INFO: Monitor thread successfully connected to server with description ServerDescription{address=localhost:27017, type=STANDALONE, state=CONNECTED, ok=true, version=ServerVersion{versionList=[5, 0, 6]}, minWireVersion=0, maxWireVersion=13, electionId=null, maxDocumentSize=16777216, roundTripTimeNanos=715100}
Sep 12, 2022 6:19:56 PM com.mongodb.diagnostics.logging.JULLoader log
INFO: Opened connection [connectionId{localValue:2, serverValue:41}] to localhost:27017
Document({_id=631f2a78ff7eda1c807a42bd, id=1, Name=Rohit, RollNo=229667, Age=35, 'website=https://www.youtube.com/results?search_query=rohit})

C:\Users\Admin\Desktop\NGT>

```

Activate Windows
Go to Settings to activate Windows.

28°C Rain 6:19 PM 9/12/2022

4. Deleting the document:

```
delete.java - Notepad
File Edit Format View Help
import com.mongodb.client.MongoCollection;
import com.mongodb.client.MongoDatabase;
import com.mongodb.MongoClient;
import com.mongodb.MongoCredential;
import org.bson.Document;
import com.mongodb.client.model.Filters;

public class delete
{
    public static void main(String args[])
    {
        MongoClient mongo = new MongoClient("localhost",27017);
        System.out.println("Connected to the database successfully");
        MongoDatabase database = mongo.getDatabase("TYITBC");
        MongoCollection<Document> collection = database.getCollection("TYITCOLL");
        System.out.println(" Collection TYICOLL selected successfully");
        collection.deleteOne(Filters.eq("id",1));
        System.out.println(" Document deleted successfully");
    }
}
```

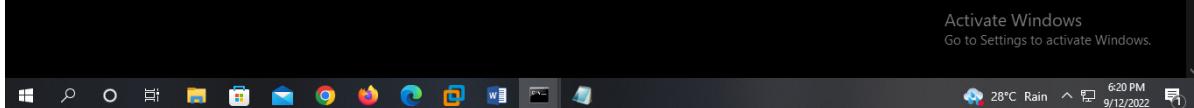


```
C:\Windows\System32\cmd.exe
Microsoft Windows [Version 10.0.19044.1889]
(c) Microsoft Corporation. All rights reserved.

C:\Users\Admin\Desktop\NGT>javac delete.java

C:\Users\Admin\Desktop\NGT>java delete
Sep 12, 2022 6:20:54 PM com.mongodb.diagnostics.logging.JULLoader log
INFO: Cluster created with settings {hosts=[localhost:27017], mode=SINGLE, requiredClusterType=UNKNOWN, serverSelectionTimeout='30000 ms', maxWaitQueueSize=500}
Connected to the database successfully
Collection TYICOLL selected successfully
Sep 12, 2022 6:20:54 PM com.mongodb.diagnostics.logging.JULLoader log
INFO: No server chosen by PrimaryServerSelector from cluster description ClusterDescription{type=UNKNOWN, connectionMode=SINGLE, all=[ServerDescription{address=localhost:27017, type=UNKNOWN, state=CONNECTING}]. Waiting for 30000 ms before timing out
Sep 12, 2022 6:20:54 PM com.mongodb.diagnostics.logging.JULLoader log
INFO: Opened connection [connectionId{localValue:1, serverValue:42}] to localhost:27017
Sep 12, 2022 6:20:54 PM com.mongodb.diagnostics.logging.JULLoader log
INFO: Monitor thread successfully connected to server with description ServerDescription{address=localhost:27017, type=STANDALONE, state=CONNECTED, ok=true, version=ServerVersion{versionList=[5, 0, 6]}, minWireVersion=0, maxWireVersion=13, electionId=null, maxDocumentSize=16777216, roundTripTimeNanos=687100}
Sep 12, 2022 6:20:54 PM com.mongodb.diagnostics.logging.JULLoader log
INFO: Opened connection [connectionId{localValue:2, serverValue:43}] to localhost:27017
Document deleted successfully

C:\Users\Admin\Desktop\NGT>
```



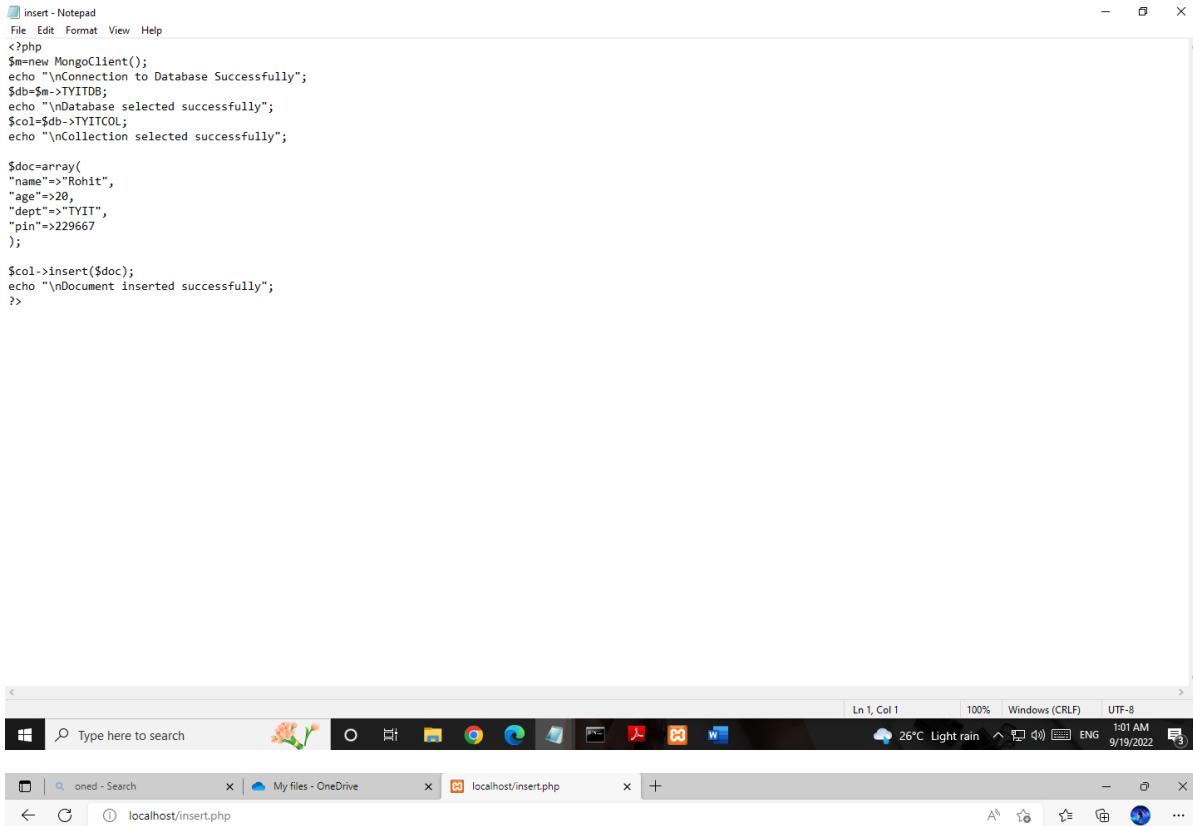
```
Command Prompt - mongo
> db.TYITCOLL.find().pretty()
>
```

Activate Windows
Go to Settings to activate Windows.



Sr No.	Date	Topic	Page No.
6		PHP and MongoDB	

1. Inserting the document:



The screenshot shows a Windows desktop environment. At the top, there is a taskbar with various icons. In the center, a Notepad window titled "insert - Notepad" displays the following PHP code:

```

<?php
$mc=new MongoClient();
echo "\nConnection to Database Successfully";
$db=$mc->TYITDB;
echo "\nDatabase selected successfully";
$col=$db->TYITCOL;
$col->insert($doc);
echo "\nCollection selected successfully";

$doc=array(
"name"=>"Rohit",
"age"=>20,
"dept"=>"TYIT",
"pin"=>229667
);

$col->insert($doc);
echo "\nDocument inserted successfully";
?>

```

Below the Notepad window, a Microsoft Edge browser window is open with the URL "localhost/insert.php". The page content is:

Connection to Database Successfully Database selected successfully Collection selected successfully Document inserted successfully

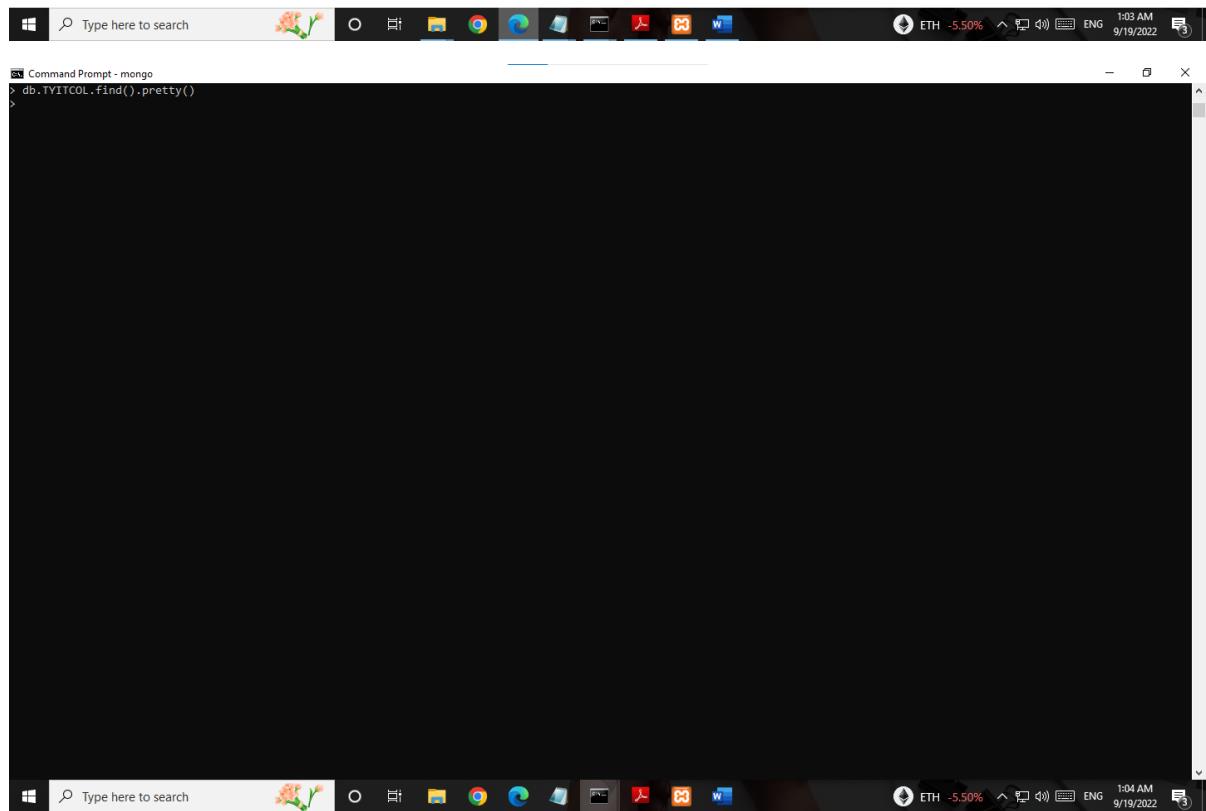
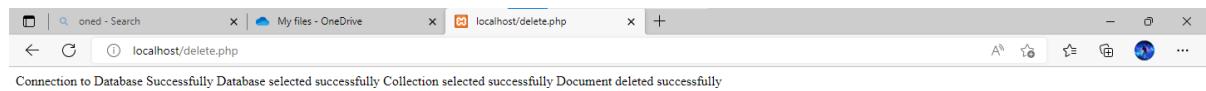


```
Command Prompt - mongo
> db.TYITCOL.find().pretty()
{
    "_id" : ObjectId("632771de7c1dfd143300002a"),
    "name" : "Rohit",
    "age" : 20,
    "dept" : "TYIT",
    "pin" : 229667
}
>
```

2. Deleting the document:

```
delete - Notepad
File Edit Format View Help
<?php
$mc=new MongoClient();
echo "\nConnection to Database Successfully";
$db=$mc->TYITDB;
echo "\nDatabase selected successfully";
$col=$db->TYITCOL;
echo "\nCollection selected successfully";

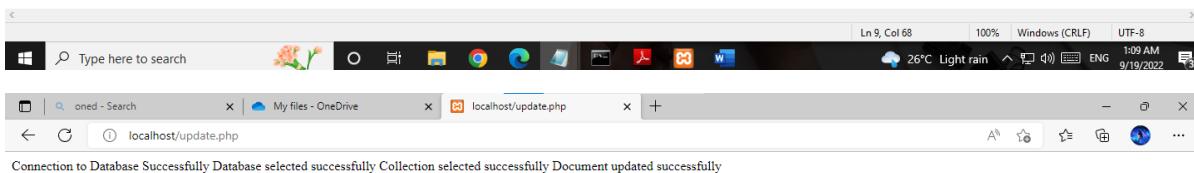
$col->remove(array("name"=>"Rohit"));
echo "\nDocument deleted successfully";
?>
```

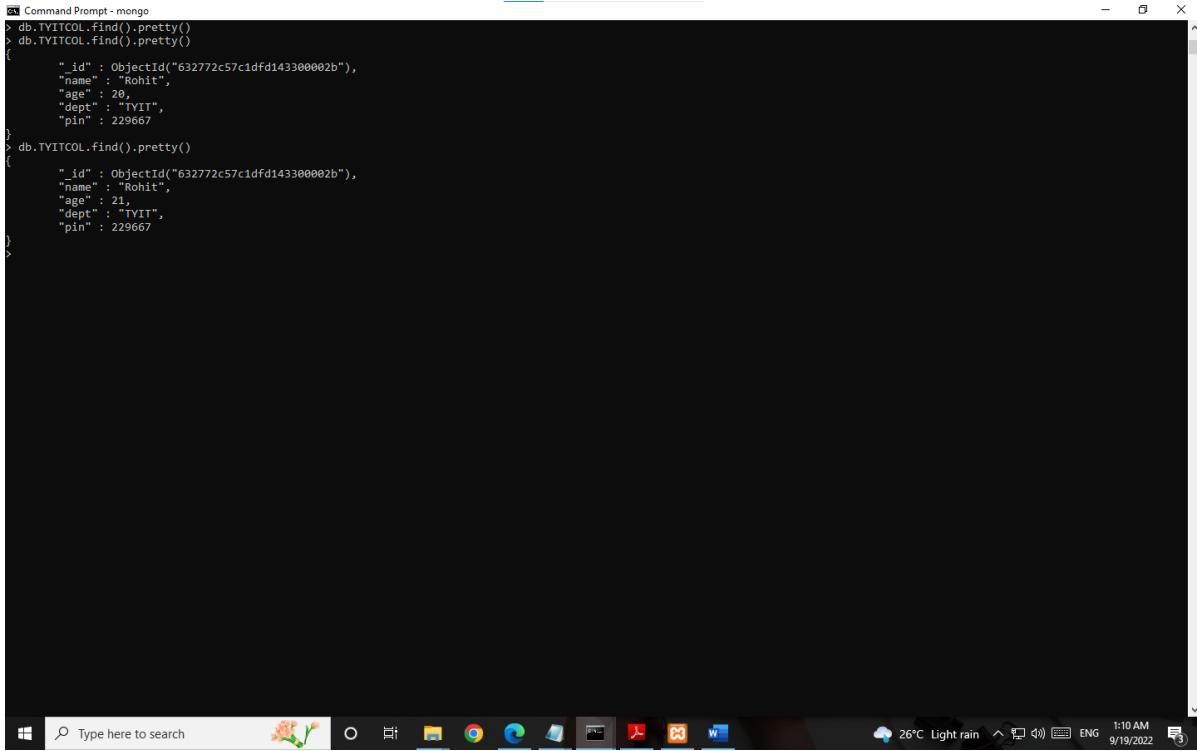


3. Updating the document:

```
update - Notepad
File Edit Format View Help
<?php
$mc=new MongoClient();
echo "\nConnection to Database Successfully";
$db=$mc->TYITDB;
echo "\nDatabase selected successfully";
$col=$db->TYITCOL;
echo "\nCollection selected successfully";

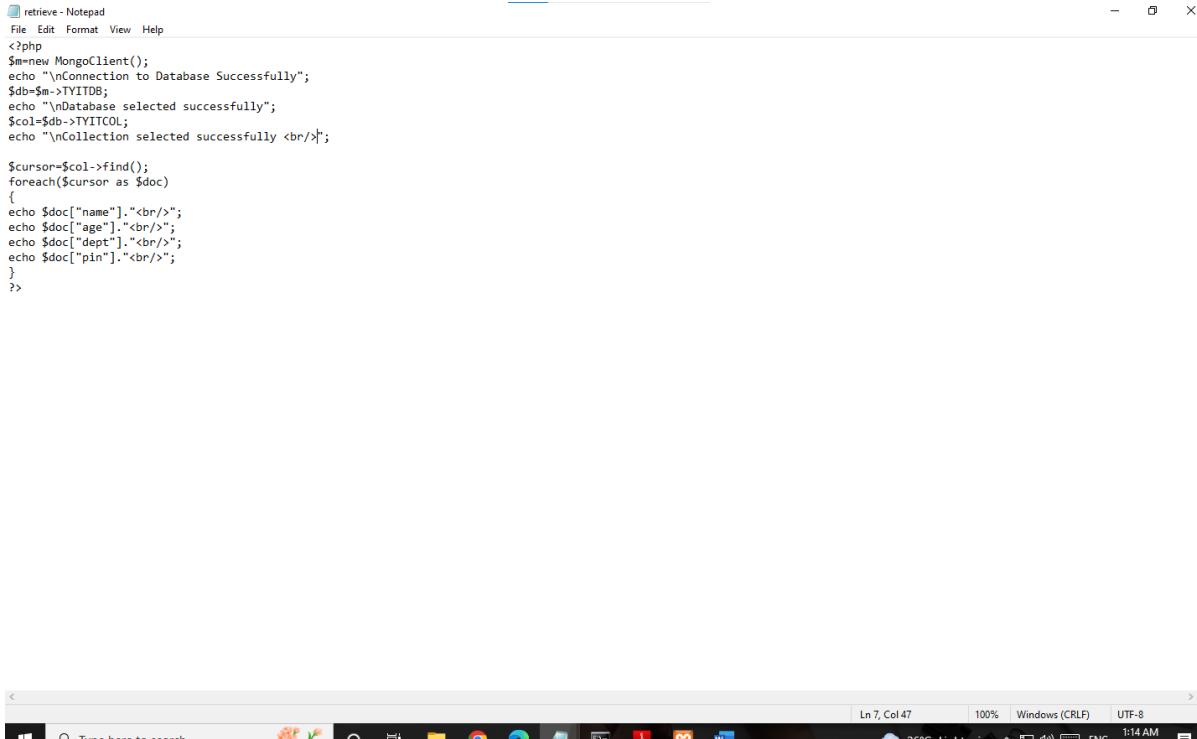
$col->update(array("name"=>"Rohit"),array('$set'=>array("age"=>21)));
echo "\nDocument updated successfully";
?>
```





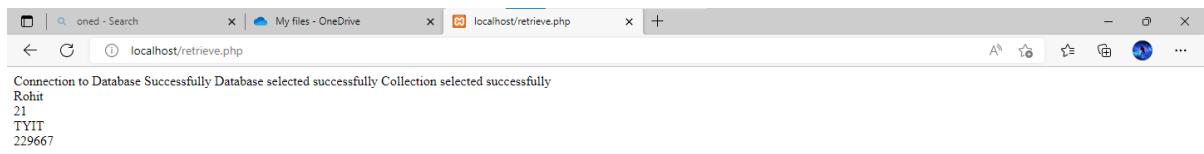
```
> db.TYITCOL.find().pretty()
> db.TYITCOL.find().pretty()
{
  "_id" : ObjectId("632772c57c1dfd143300002b"),
  "name" : "Rohit",
  "age" : 20,
  "dept" : "TYIT",
  "pin" : 229667
}
> db.TYITCOL.find().pretty()
{
  "_id" : ObjectId("632772c57c1dfd143300002b"),
  "name" : "Rohit",
  "age" : 21,
  "dept" : "TYIT",
  "pin" : 229667
}
>
```

4. Retrieving the document:



```
retrieve - Notepad
File Edit Format View Help
<?php
$mc=new MongoClient();
echo "\nConnection to Database Successfully";
$db=$mc->TYITDB;
echo "\nDatabase selected successfully";
$col=$db->TYITCOL;
echo "\nCollection selected successfully <br/>";

$cursor=$col->find();
foreach($cursor as $doc)
{
  echo $doc["name"]."<br/>";
  echo $doc["age"]."<br/>";
  echo $doc["dept"]."<br/>";
  echo $doc["pin"]."<br/>";
}
?>
```



Sr No.	Date	Topic	Page No.
7		Python and MongoDB	

1. Inserting the document:

```

insert.py - C:\Users\Tron\Desktop\NGT\insert.py (3.10.5)
File Edit Format Run Options Window Help
from pymongo import MongoClient
client=MongoClient('localhost',27017)
db=client.TYITDB
def insert():
    try:
        nl=input("Enter the name : ")
        al=int(input("Enter the age : "))
        dl=input("Enter the department : ")
        pl=int(input("Enter the pin no. : "))
        db.TYITCOL.insert_one(
            {
                "name" : nl,
                "age" : al,
                "department" : dl,
                "pin_no" : pl
            })
        print("Inserted Successfully")
    except Exception:
        print(str(e))
    insert()

```

```

IDLE Shell 3.10.5
File Edit Shell Debug Options Window Help
Python 3.10.5 (tags/v3.10.5:f377153, Jun  6 2022, 16:14:13) [MSC v.1929 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>> ===== RESTART: C:\Users\Tron\Desktop\NGT\insert.py =====
Enter the name : Rohit
Enter the age : 20
Enter the department : TYIT
Enter the pin no. : 229667
Inserted Successfully
>>>

```

```
Command Prompt - mongo
> db.TYITCOL.find().pretty()
{
    "_id" : ObjectId("63275b524def061364c1d389"),
    "name" : "Rohit",
    "age" : 20,
    "department" : "TYIT",
    "pin_no" : 229667
}
>
```

2. Updating the document:

```
update.py - C:/Users/Tron/Desktop/NGT/update.py (3.10.5)
File Edit Format Run Options Window Help
from pymongo import MongoClient
client=MongoClient('localhost',27017)
db=client.TYITDB
def update():
    try:
        n1=input("Enter the name : ")
        a1=int(input("Enter the updated age : "))
        db.TYITCOL.update_one(
            {"name":n1}, {"$set":{"age":a1}})
        print("Updated successfully")
    except Exception:
        print(str(e))
update()
```

The screenshot shows the Python IDLE Shell interface. The code in the shell window is as follows:

```
>>> ===== RESTART: C:/Users/Tron/Desktop/NGT/insert.py =====
Enter the name : Rohit
Enter the age : 20
Enter the department : TYIT
Enter the pin_no : 229667
Inserted Successfully
>>> ===== RESTART: C:/Users/Tron/Desktop/NGT/update.py =====
Enter the name : Rohit
Enter the updated age : 21
Traceback (most recent call last):
  File "C:/Users/Tron/Desktop/NGT/update.py", line 9, in update
    db.TYITCOL.update(
      File "C:/Users/Tron/AppData/Local/Programs/Python/310/lib/site-packages/pymongo/collection.py", line 3200, in __call__
        raise TypeError(
TypeError: 'Collection' object is not callable. If you meant to call the 'update' method on a 'Collection' object it is failing because no such method exists.

During handling of the above exception, another exception occurred:

Traceback (most recent call last):
  File "C:/Users/Tron/Desktop/NGT/update.py", line 14, in <module>
    update()
  File "C:/Users/Tron/Desktop/NGT/update.py", line 13, in update
    print(str(e))
NameError: name 'e' is not defined
>>> ===== RESTART: C:/Users/Tron/Desktop/NGT/update.py =====
Enter the name : Rohit
Enter the updated age : 21
Updated successfully
>>> |
```

The screenshot shows the Windows Command Prompt window with the title 'Command Prompt - mongo'. The commands entered are:

```
> db.TYITCOL.find().pretty()
{
  "_id" : ObjectId("63275b524def061364c1d389"),
  "name" : "Rohit",
  "age" : 20,
  "department" : "TYIT",
  "pin_no" : 229667
}
> db.TYITCOL.find().pretty()
{
  "_id" : ObjectId("63275b524def061364c1d389"),
  "name" : "Rohit",
  "age" : 21,
  "department" : "TYIT",
  "pin_no" : 229667
}
```

3. Retrieving data from the document:

ret.py - C:/Users/Tron/Desktop/NGT/ret.py (3.10.5)

```
File Edit Format Run Options Window Help
from pymongo import MongoClient
client=MongoClient('localhost',27017)
db=client.TYITDB

def ret():
    try:
        read= db.TYITCOL.find()
        print("data")
        for one in read:
            print(one)
    except Exception:
        print(str(e))
ret()
```

IDLE Shell 3.10.5

```
File Edit Shell Debug Options Window Help
Python 3.10.5 (tags/v3.10.5:f377153, Jun  6 2022, 16:14:13) [MSC v.1929 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license()" for more information.

>>> ===== RESTART: C:/Users/Tron/Desktop/NGT/insert.py =====
Enter the name : Rohit
Enter the age : 20
Enter the department : TYIT
Enter the pin no. : 229667
Inserted Successfully

>>> ===== RESTART: C:/Users/Tron/Desktop/NGT/update.py =====
Enter the name : Rohit
Enter the updated age : 21
Traceback (most recent call last):
  File "C:/Users/Tron/Desktop/NGT/update.py", line 9, in update
    db.TYITCOL.update(
      File "C:/Users/Tron/AppData\Local\Programs\Python\Python310\lib\site-packages\pymongo\collection.py", line 3200, in __call__
        raise TypeError(
TypeError: 'Collection' object is not callable. If you meant to call the 'update' method on a 'Collection' object it is failing because no such method exists.

During handling of the above exception, another exception occurred:

Traceback (most recent call last):
  File "C:/Users/Tron/Desktop/NGT/update.py", line 14, in <module>
    update()
  | File "C:/Users/Tron/Desktop/NGT/update.py", line 13, in update
    print(str(e))
NameError: name 'e' is not defined

>>> ===== RESTART: C:/Users/Tron/Desktop/NGT/update.py =====
Enter the name : Rohit
Enter the updated age : 21
Updated successfully

>>> ===== RESTART: C:/Users/Tron/Desktop/NGT/ret.py =====
data
{'_id': ObjectId('63275b524def061364c1d389'), 'name': 'Rohit', 'age': 21, 'department': 'TYIT', 'pin_no': 229667}
```

4. Deleting the document:

```
delete.py - C:/Users/Tron/Desktop/NGT/delete.py (3.10.5)
File Edit Format Run Options Window Help
from pymongo import MongoClient
client = MongoClient('localhost', 27017)
db=client.TYITDB

def delete():
    try:
        n1=input("Enter the name : ")
        db.TYITCOL.delete_one(
            {"name": n1})
        print("Deleted Successfully")
    except Exception:
        print(str(e))
delete()
|
```

```
IDLE Shell 3.10.5
File Edit Shell Debug Options Window Help
Python 3.10.5 (tags/v3.10.5:f377153, Jun  6 2022, 16:14:13) [MSC v.1929 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>> ===== RESTART: C:/Users/Tron/Desktop/NGT/insert.py =====
Enter the name : Rohit
Enter the age : 20
Enter the department : TYIT
Enter the pin no. : 229667
Inserted Successfully
>>> ===== RESTART: C:/Users/Tron/Desktop/NGT/update.py =====
Enter the name : Rohit
Enter the updated age : 21
Traceback (most recent call last):
  File "C:/Users/Tron/Desktop/NGT/update.py", line 9, in update
    db.TYITCOL.update(
      File "C:/Users/Tron/AppData/Local/Programs/Python/3.10/lib/site-packages/pymongo/collection.py", line 3200, in __call__
        raise TypeError(
TypeError: 'Collection' object is not callable. If you meant to call the 'update' method on a 'Collection' object it is failing because no such method exists.

During handling of the above exception, another exception occurred:

Traceback (most recent call last):
  File "C:/Users/Tron/Desktop/NGT/update.py", line 14, in <module>
    update()
  File "C:/Users/Tron/Desktop/NGT/update.py", line 13, in update
    print(str(e))
NameError: name 'e' is not defined
>>> ===== RESTART: C:/Users/Tron/Desktop/NGT/update.py =====
Enter the name : Rohit
Enter the updated age : 21
Updated successfully
>>> ===== RESTART: C:/Users/Tron/Desktop/NGT/ret.py =====
data
({_id: ObjectId('63275b524def061364c1d389'), 'name': 'Rohit', 'age': 21, 'department': 'TYIT', 'pin_no': 229667})
>>> ===== RESTART: C:/Users/Tron/Desktop/NGT/delete.py =====
Enter the name : Rohit
Deleted Successfully
>>>
```

```
Command Prompt - mongo
> db.TYITCOL.find().pretty()
{
    "_id" : ObjectId("63275b524def061364c1d389"),
    "name" : "Rohit",
    "age" : 20,
    "department" : "TYIT",
    "pin_no" : 229667
}
> db.TYITCOL.find().pretty()
{
    "_id" : ObjectId("63275b524def061364c1d389"),
    "name" : "Rohit",
    "age" : 21,
    "department" : "TYIT",
    "pin_no" : 229667
}
> db.TYITCOL.find().pretty()
>
```

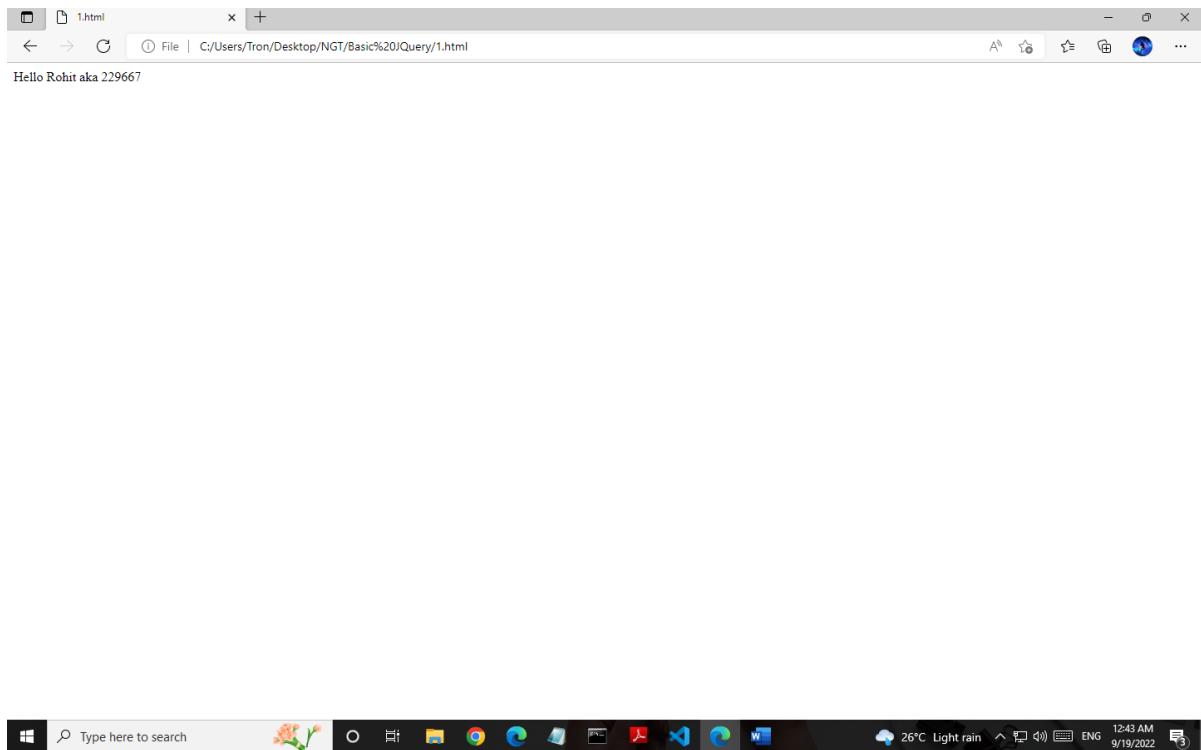


Sr No.	Date	Topic	Page No.
8		Programs on Basic jQuery	

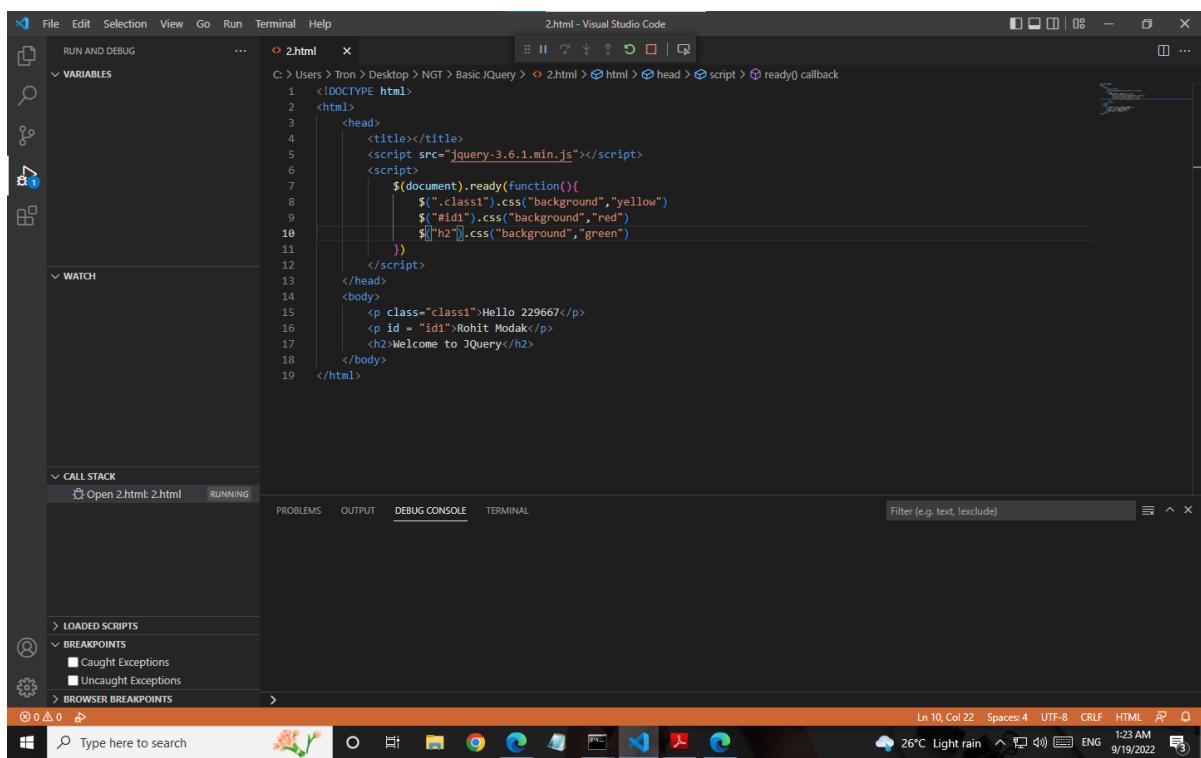
1. Write a jQuery to check the content of the element on button click:

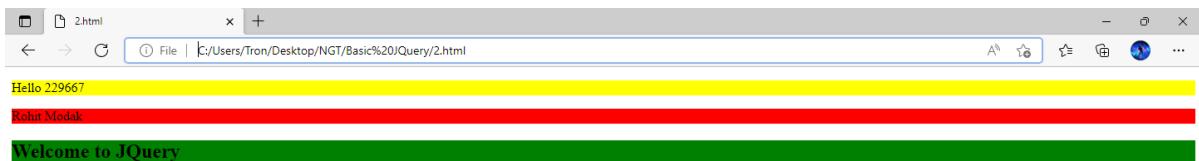
The screenshot shows the Visual Studio Code interface with the following details:

- Editor Area:** Displays the code for "1.html". The code includes a script block that uses jQuery to write "Hello Rohit aka 229667" to the page when a button is clicked.
- Run and Debug View:** Shows the current file "1.html" is selected.
- VARIABLES View:** Empty.
- WATCH View:** Empty.
- CALL STACK View:** Shows an uncaught ReferenceError: \$ is not defined at <anonymous> (c:\Users\Tron\Desktop\NGT\Basic JQuery\1.html:7:13).
- Bottom Status Bar:** Shows the file path C:/Users/Tron/Desktop/NGT/Basic%20JQuery/1.html, line 18, column 8, and other system information like weather and time.
- Browser Preview:** A separate window shows the rendered HTML with a "Welcome to PHP" heading and a "Click Me" button.



2. Write a jQuery to select element by class name, id and element name:



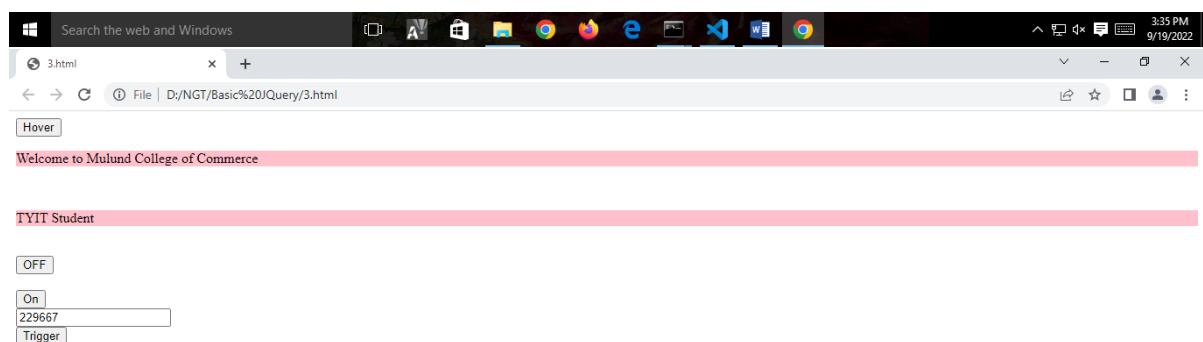


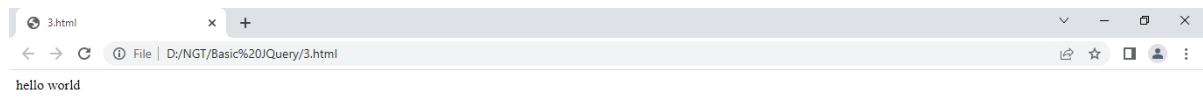
3. Write a jQuery to show the use of click (), hover (), on (), trigger (), off () event:

A screenshot of Visual Studio Code showing a file named "3.html". The code uses jQuery to demonstrate various events:

```
1 <!DOCTYPE html>
2 <html>
3   <head>
4     <title></title>
5     <script src="jquery-3.6.1.min.js"></script>
6     <script>
7       $(document).ready(function(){
8         $("*").hover(function(){
9           document.write("hello world")
10        });
11        $("p").on("click", function(){
12          $(this).css("background-color", "pink")
13        });
14        $("#b2").click(function(){
15          $("p").off("click")
16        });
17        $("#b3").on("click",function(){
18          $("#i1").hide()
19        });
20        $("input").select(function(){
21          $("input").after("Text marked!")
22        });
23        $("#b4").click(function(){
24          $("input").trigger("select")
25        });
26      });
27    </script>
28  </head>
29  <body>
30    <button id="b1">Hover</button><br>
31    <p>Welcome to Mulund college of commerce</p><br>
32    <p>TNT Student</p><br>
33    <button id="b2">OFF</button><br>
34    <p id="t1">Hello All</p><br>
35    <button id="b3">On</button><br>
36    <input type="text" value="229667"><br>
37    <button id="b4">Trigger</button>
38  </body>
39 </html>
```

The code includes functions for hovering over elements, clicking on paragraphs, selecting inputs, and triggering a select event on buttons. It also demonstrates the use of the document ready function and event delegation.





4. Write a jQuery to create animated hide effect:

File Edit Selection View Go Run Terminal Help

4.html - Visual Studio Code

RUN AND DEBUG

VARIABLES

WATCH

CALL STACK

Open 4.html: 4.html RUNNING

LOADED SCRIPTS

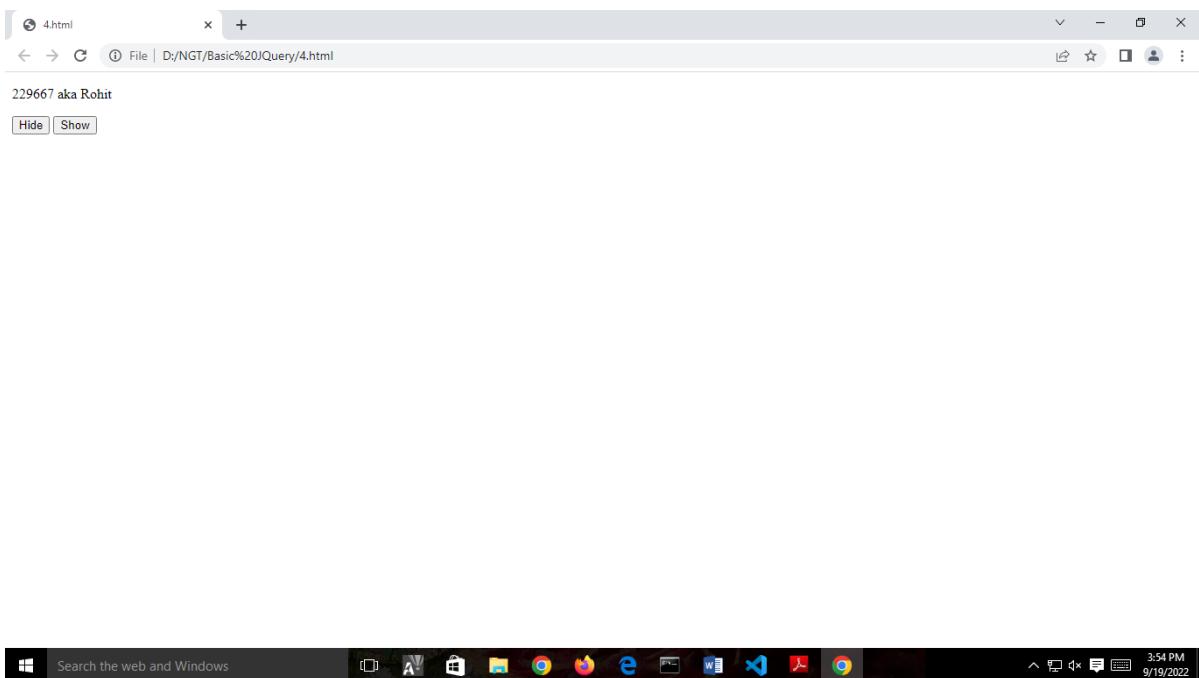
BREAKPOINTS

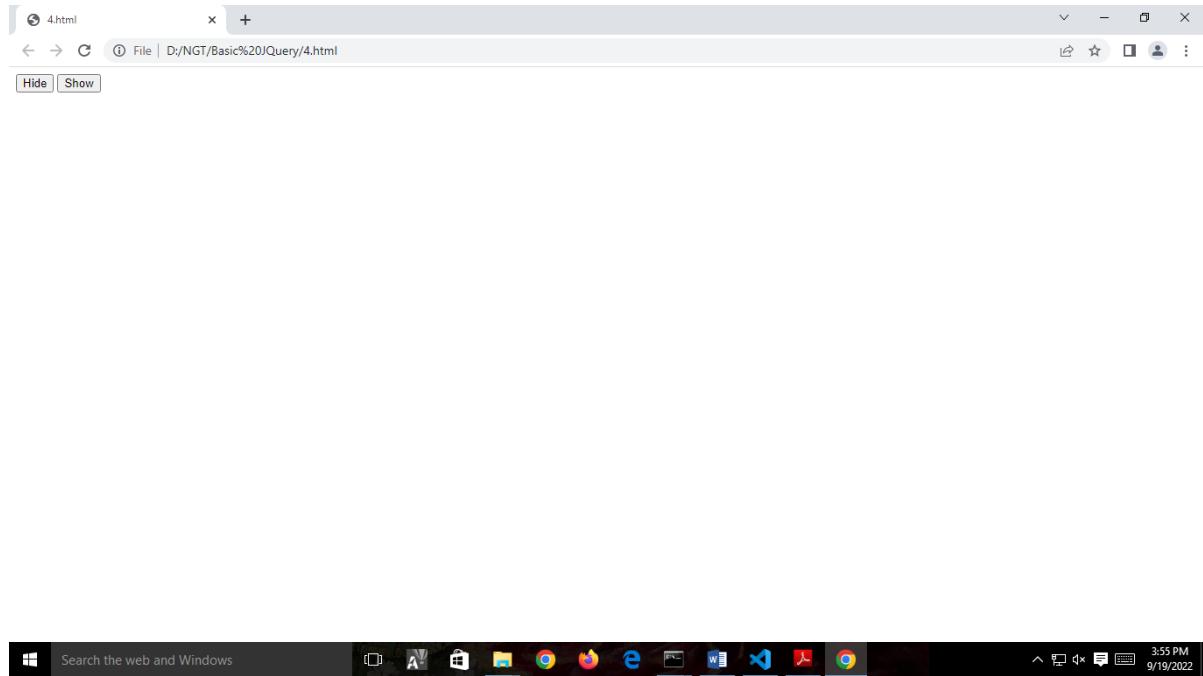
- Caught Exceptions
- Uncought Exceptions

BROWSER BREAKPOINTS

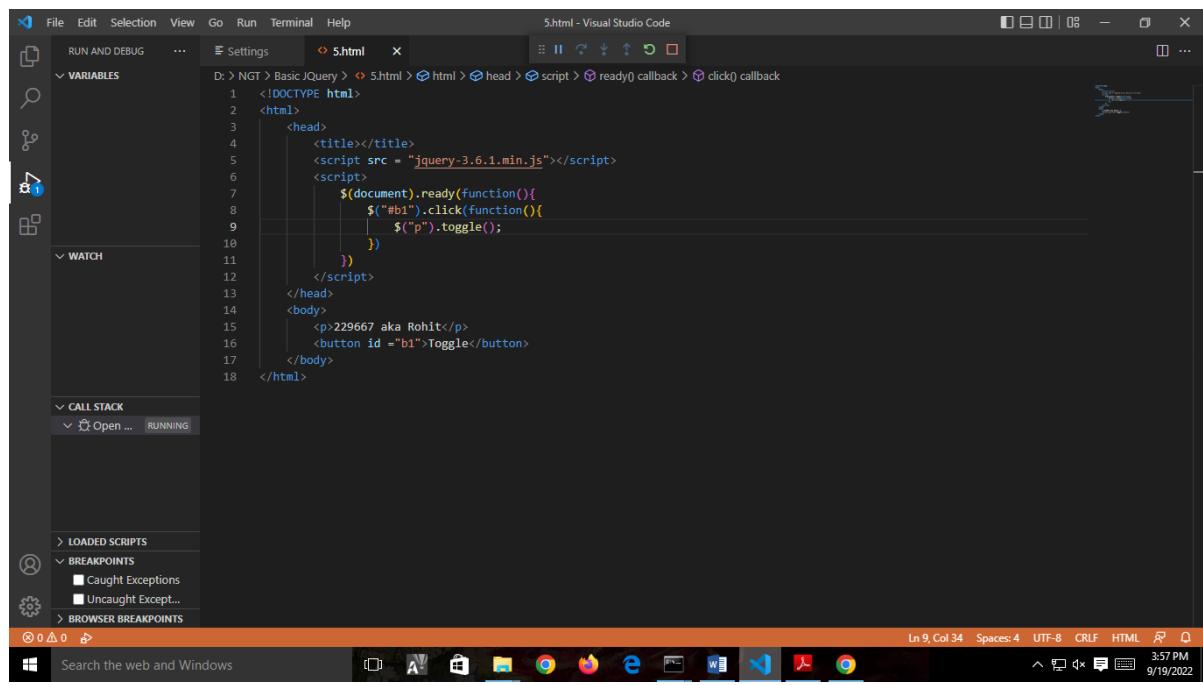
Ln 7, Col 42 Spaces: 4 UTF-8 CRLF HTML ⚡ 3:54 PM 9/19/2022

```
D:\NGT>Basic%20jQuery> 4.html > html > head > script > ready0 callback
1  <!DOCTYPE html>
2  <html>
3  <head>
4  <title></title>
5  <script src = "jquery-3.6.1.min.js"></script>
6  <script>
7  $(document).ready(function(){
8      $("#b1").click(function(){
9          $("p").hide();
10     });
11     $("#b2").click(function(){
12         $("p").show();
13     });
14 });
15 </script>
16 </head>
17 <body>
18 <p>229667 aka Rohit</p>
19 <button id = "b1">Hide</button>
20 <button id = "b2">Show</button>
21 </body>
22 </html>
```





5. Write a jQuery to create a simple toggle effect:

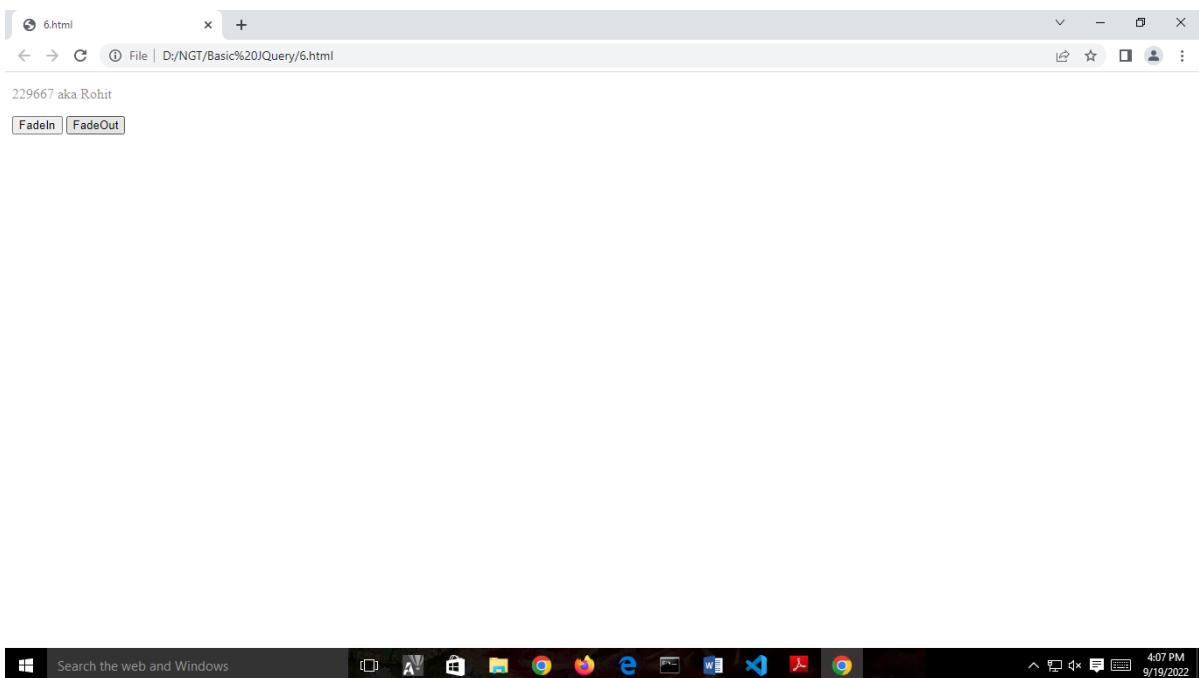




6. Write a jQuery to create fade-in and fade-out effects:

The screenshot shows the Visual Studio Code interface in dark mode. The main area displays a file named '6.html' containing a basic jQuery script. The script includes a reference to 'jquery-3.6.1.min.js' and uses jQuery's ready() and click() methods to fade in and out a paragraph element. The left sidebar features sections for 'RUN AND DEBUG', 'VARIABLES', 'WATCH', and 'CALL STACK'. The bottom status bar shows the file path 'D:\NGT\Basic%20jQuery\6.html', line 'Ln 12, Col 40', and encoding 'UTF-8'. The taskbar at the bottom includes icons for various Windows applications like File Explorer, Task View, and Edge.

```
<!DOCTYPE html>
<html>
<head>
    <title></title>
    <script src = "jquery-3.6.1.min.js"></script>
    <script>
        $(document).ready(function(){
            $("#b1").click(function(){
                $("p").fadeIn(2000);
            });
            $("#b2").click(function(){
                $("p").fadeOut(2500);
            });
        })
    </script>
</head>
<body>
    <p>229667 aka Rohit</p>
    <button id ="b1">FadeIn</button>
    <button id ="b2">FadeOut</button>
</body>
</html>
```

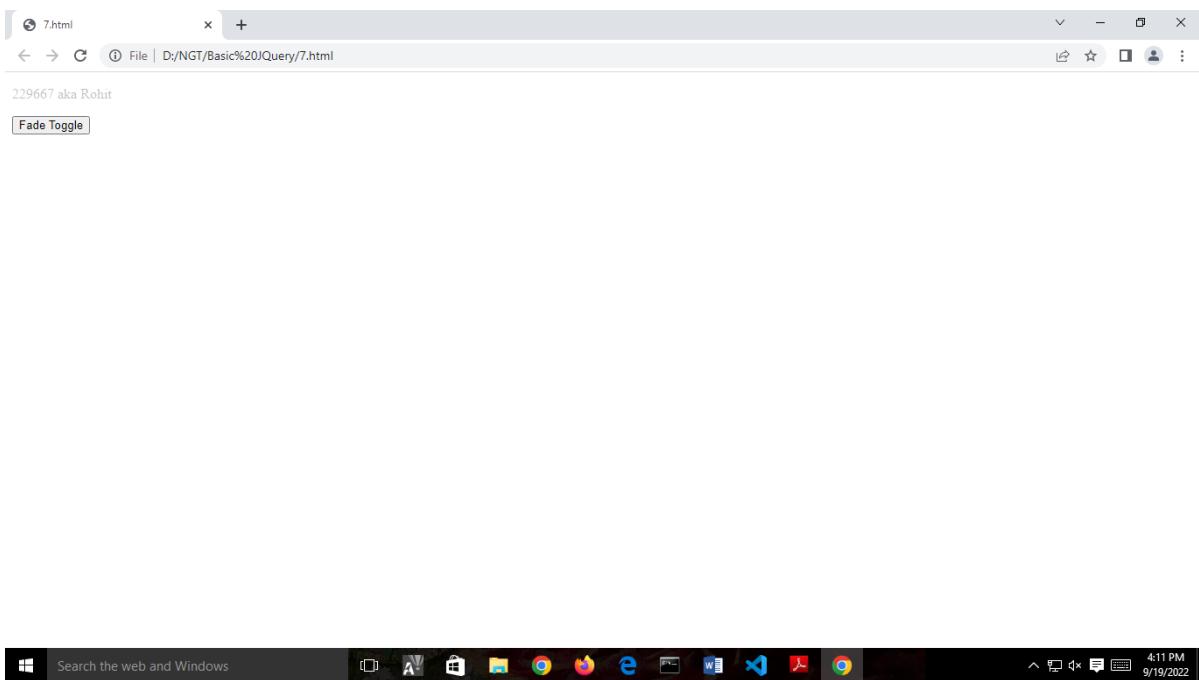


7. Write a jQuery to create fade toggle effects:

The screenshot shows the Visual Studio Code interface. The code editor displays a file named "7.html" containing the following jQuery script:

```
<!DOCTYPE html>
<html>
  <head>
    <title></title>
    <script src = "jquery-3.6.1.min.js"></script>
    <script>
      $(document).ready(function(){
        $("#b1").click(function(){
          $("p").fadeToggle();
        })
      })
    </script>
  </head>
  <body>
    <p>229667 aka Rohit</p>
    <button id ="b1">Fade Toggle</button>
  </body>
</html>
```

The run and debug sidebar on the left shows variables, watch, and call stack sections. The bottom status bar indicates the file is "7.html - Visual Studio Code", has 9 lines and 38 columns, uses spaces for indentation, and is in UTF-8 encoding.

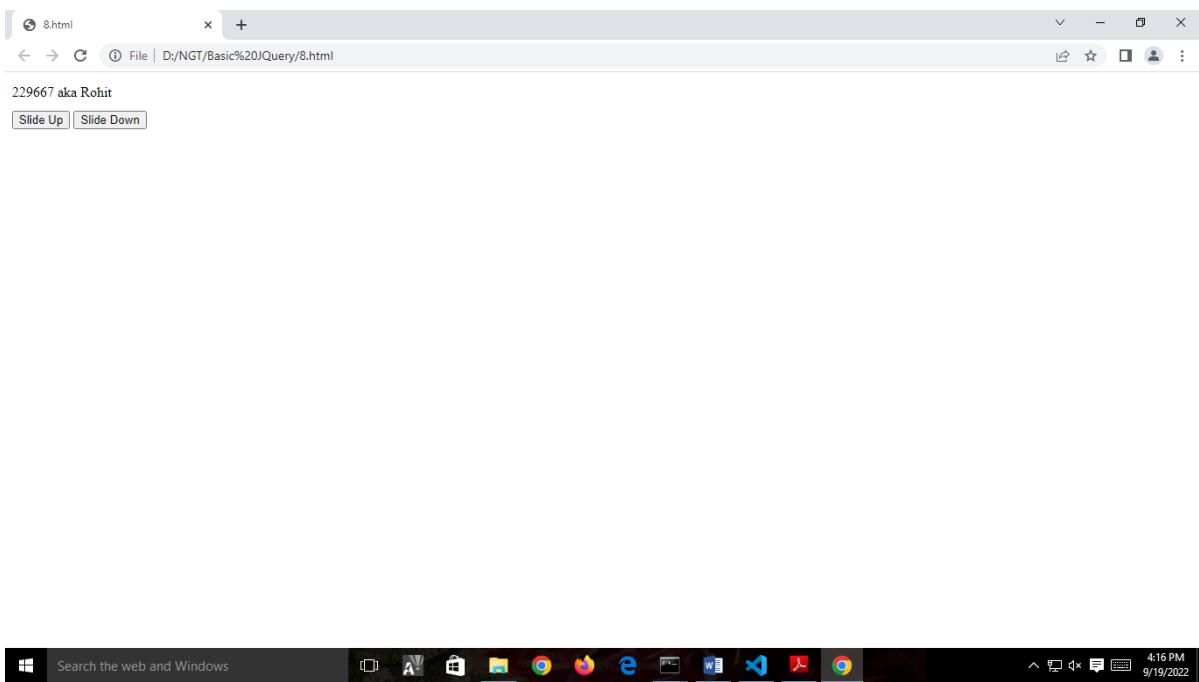


8. Write a jQuery to create a slide-up and slide-down effects:

The screenshot shows the Visual Studio Code interface with the following details:

- File Bar:** File, Edit, Selection, View, Go, Run, Terminal, Help.
- Title Bar:** 8.html - Visual Studio Code.
- Left Sidebar:**
 - RUN AND DEBUG
 - VARIABLES
 - WATCH
 - CALL STACK
 - LOADED SCRIPTS
 - BREAKPOINTS
 - Caught Exceptions
 - Unc caught Except...
 - BROWSER BREAKPOINTS
- Code Editor:** Displays the following jQuery script (lines 1-22):

```
1  <!DOCTYPE html>
2  <html>
3      <head>
4          <title></title>
5          <script src = "jquery-3.6.1.min.js"></script>
6          <script>
7              $(document).ready(function(){
8                  $("#b1").click(function(){
9                      $("p").slideUp(5000);
10                 });
11                 $("#b2").click(function(){
12                     $("p").slideDown(5000);
13                 });
14             });
15         </script>
16     </head>
17     <body>
18         <p>229667 aka Rohit</p>
19         <button id ="b1">Slide Up</button>
20         <button id = "b2">Slide Down</button>
21     </body>
22 </html>
```

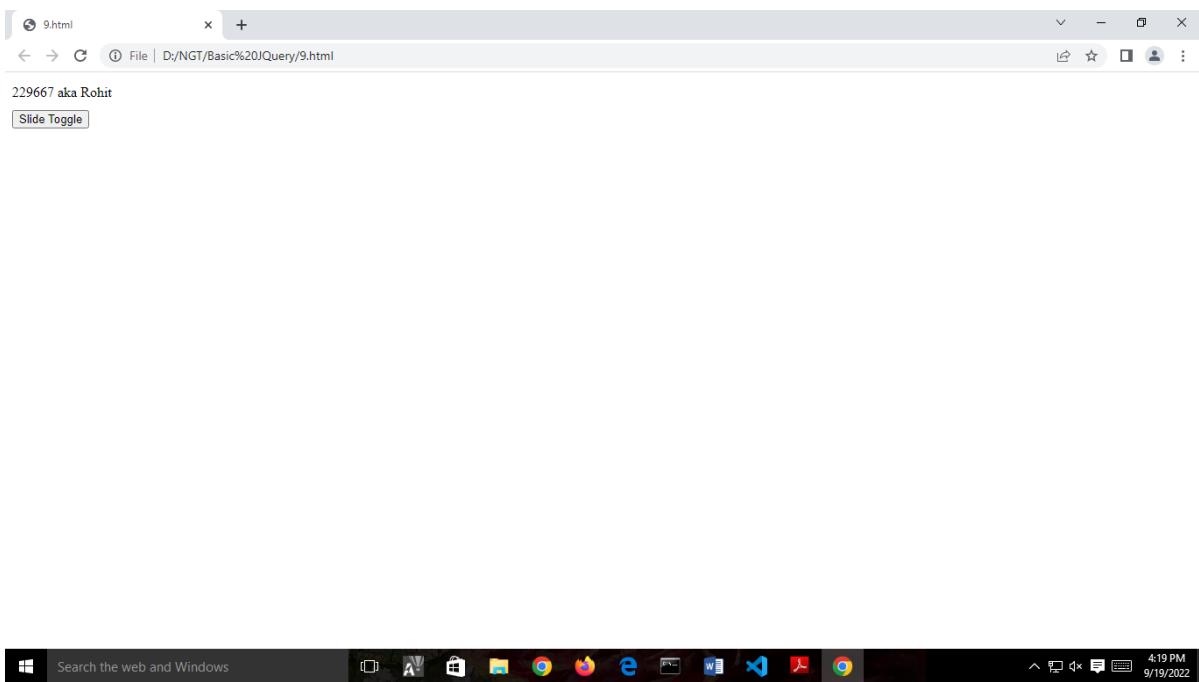
- Bottom Status Bar:** Line 12, Col 42, Spaces: 4, UTF-8, CRLF, HTML, 4:16 PM, 9/19/2022.


9. Write a jQuery to create slide toggle effects:

The screenshot shows the Visual Studio Code interface. The left sidebar has icons for file operations, search, and run/debug. The main area is titled "9.html - Visual Studio Code". The code editor displays the following HTML and JavaScript code:

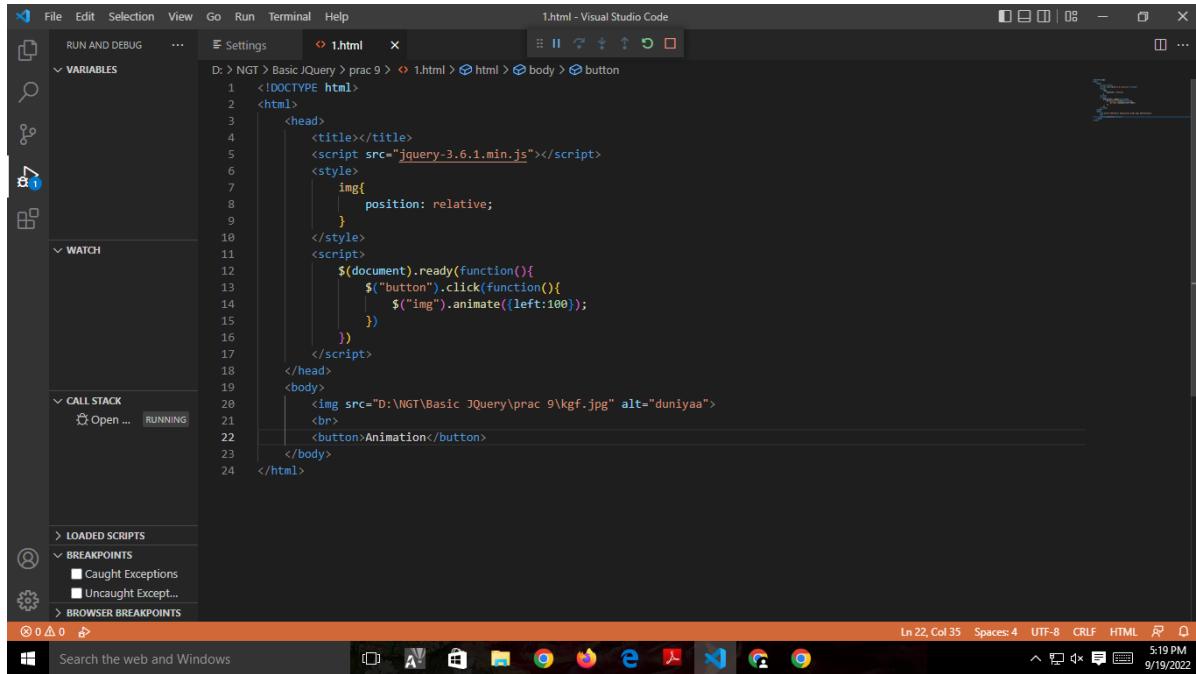
```
D: > NGT > Basic JQuery > 9.html > html > head > script > ready() callback > click() callback
1  <!DOCTYPE html>
2  <html>
3      <head>
4          <title></title>
5          <script src = "jquery-3.6.1.min.js"></script>
6          <script>
7              $(document).ready(function(){
8                  $("#b1").click(function(){
9                      $("p").slideToggle();
10                 })
11             })
12         </script>
13     </head>
14     <body>
15         <p>229667 aka Rohit</p>
16         <button id ="b1">Slide Toggle</button>
17     </body>
18 </html>
```

The status bar at the bottom shows "Ln 9, Col 39" and "4:20 PM 9/19/2022".



Sr No.	Date	Topic	Page No.
9		jQuery Advanced	

1. Write a jQuery to create Animation effects:



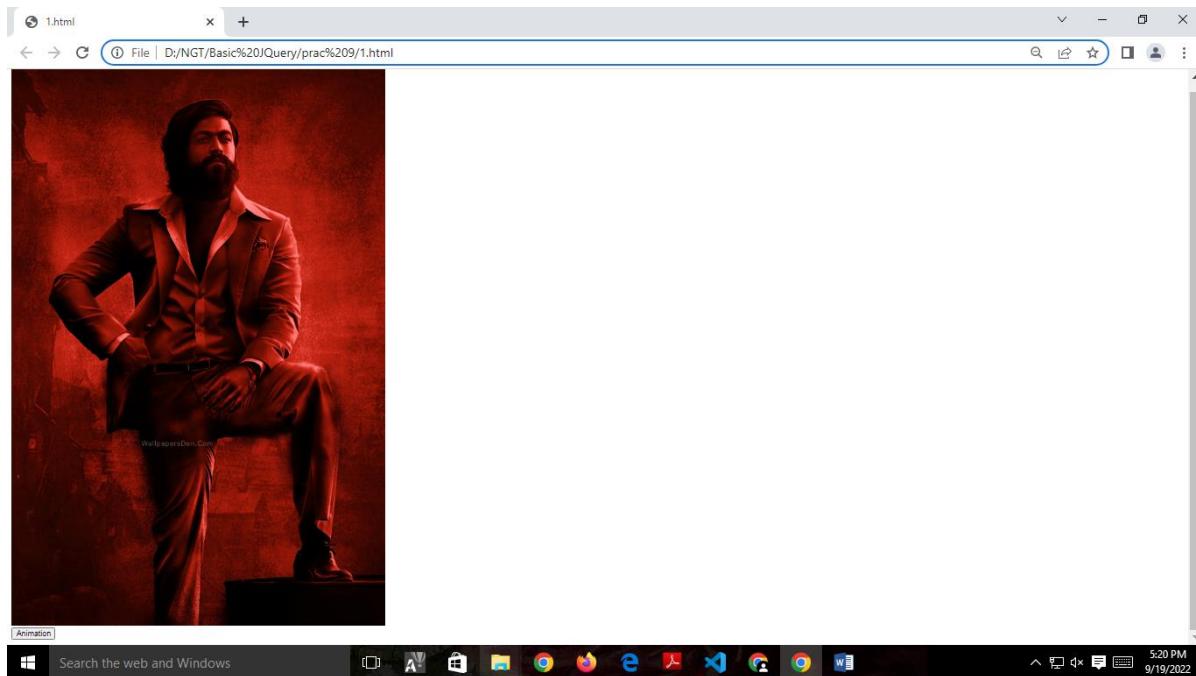
The screenshot shows the Visual Studio Code interface with the following details:

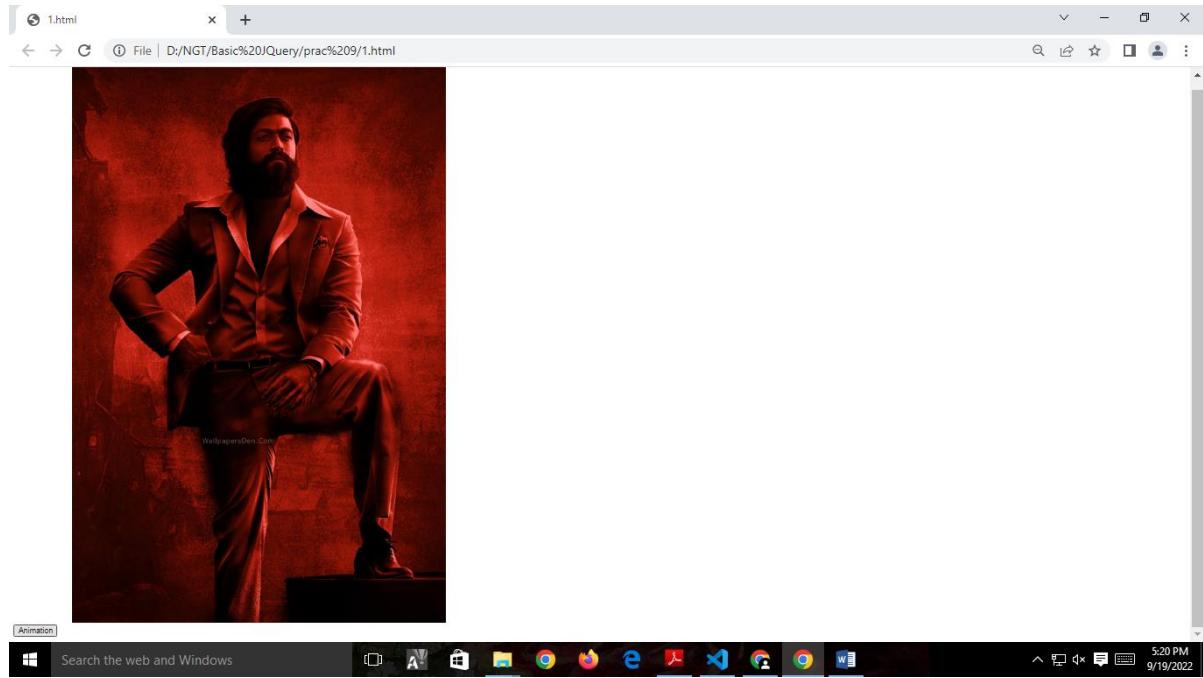
- File Path:** D:\NGT\Basic JQuery\prac 9\1.html
- Code Content:**

```

1 <!DOCTYPE html>
2 <html>
3   <head>
4     <title></title>
5     <script src="jquery-3.6.1.min.js"></script>
6     <style>
7       img{
8         position: relative;
9       }
10      </style>
11      <script>
12        $(document).ready(function(){
13          $("button").click(function(){
14            $("img").animate({left:100});
15          })
16        })
17      </script>
18    </head>
19    <body>
20      
21      <br>
22      <button>Animation</button>
23    </body>
24  </html>

```
- Toolbars and Side Panels:**
 - VARIABLES**: Shows the current variable state.
 - WATCH**: Shows the current watch list.
 - CALL STACK**: Shows the current call stack.
 - LOADED SCRIPTS**: Shows the loaded scripts.
 - BREAKPOINTS**: Shows caught and uncaught exceptions.
 - ROWSER BREAKPOINTS**: Shows browser breakpoints.
- Status Bar:** Shows the line number (Ln 22, Col 35), spaces (Spaces: 4), encoding (UTF-8), file type (HTML), and date/time (5:19 PM 9/19/2022).



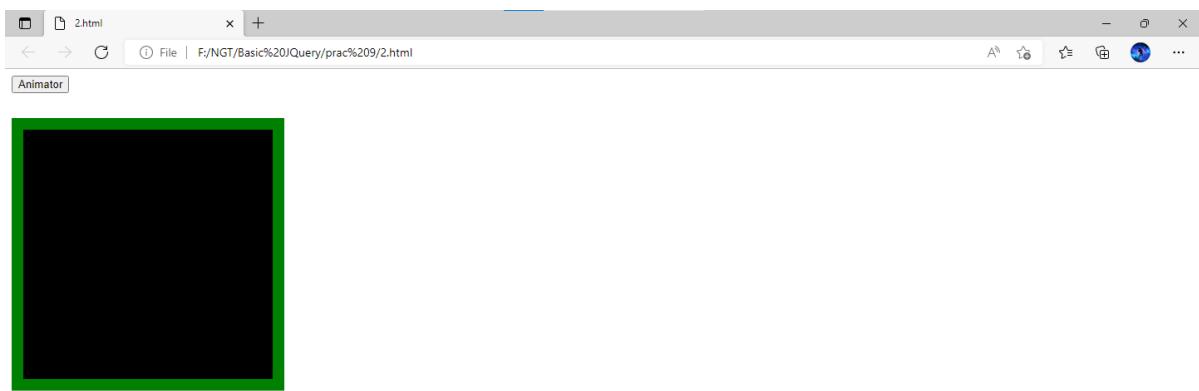
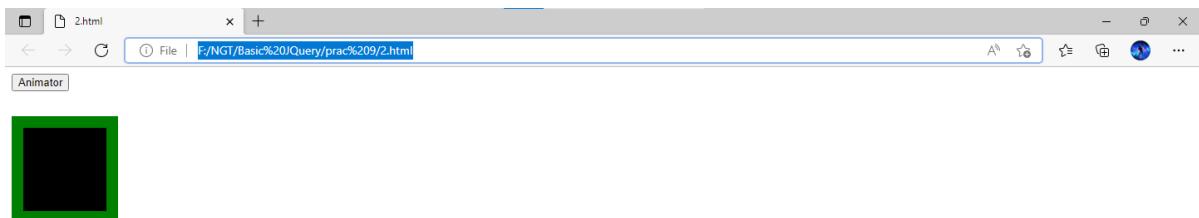


2. Write a jQuery animate multiple CSS properties:

A screenshot of Visual Studio Code showing the code for a jQuery animation. The file is named "2.html". The code includes a CSS style for ".box" and a JavaScript function that uses jQuery's .animate() method to change the width and height of the ".box" element when a button is clicked.

```
RUN AND DEBUG ...  
RUN  
Run and Debug  
To customize Run and Debug, open a folder and create a launch.json file.  
Show all automatic debug configurations.  
2.html x  
F: > NGT > Basic jQuery > prac 9 > 2.html > html > body > div.box  
1 <!DOCTYPE html>  
2 <html>  
3 <head>  
4 <script src="jquery-3.6.1.min.js"></script>  
5 <style type="text/css">  
6 .box{  
7     width:100px;  
8     height:100px;  
9     background: black;  
10    margin-top:30px;  
11    border-style: solid;  
12    border-color: green;  
13    border-width: 14px;  
14 }  
15 </style>  
16 <script>  
17     $(document).ready(function(){  
18         $("button").click(function(){  
19             $(".box").animate({  
20                 width:"300px",  
21                 height:"300px"  
22             })  
23         })  
24     })  
25 </script>  
26 </head>  
27 <body>  
28     <button>Animator</button>  
29     <div class = "box"></div>  
30 </body>  
31 </html>
```

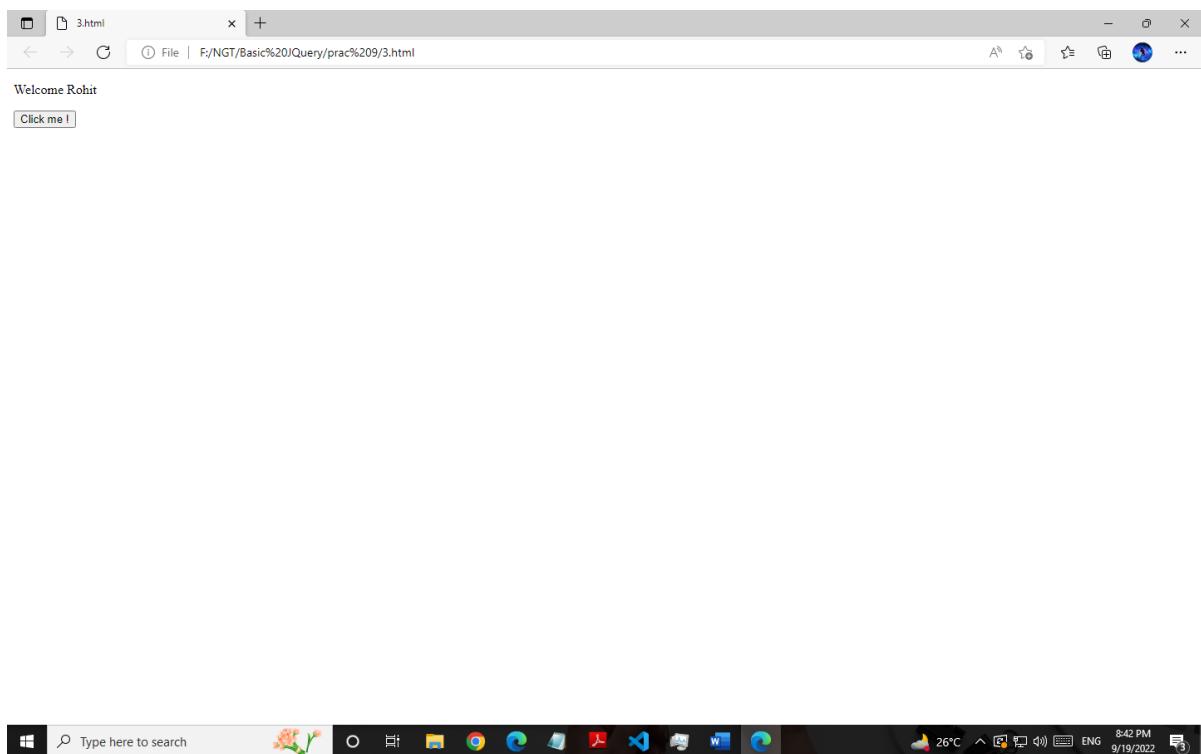
The Visual Studio Code interface includes a sidebar for "RUN AND DEBUG", a status bar at the bottom, and a taskbar at the very bottom.

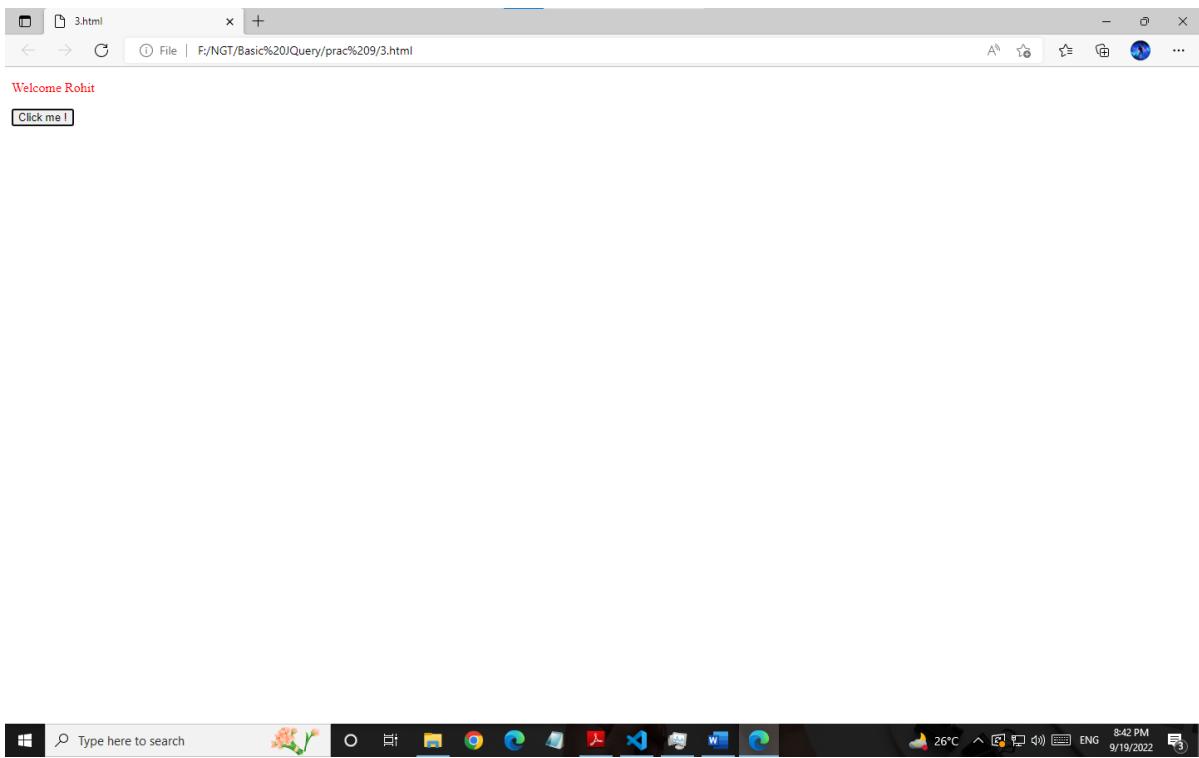


3. Write a jQuery to perform method of chaining:

A screenshot of the Visual Studio Code interface. The main area shows the code for a file named '3.html'. The code includes a script that changes the color of a paragraph to red when a button is clicked. The interface includes a 'RUN AND DEBUG' panel, a 'WATCH' panel, and a 'CALL STACK' panel. The bottom status bar shows the file path 'F:\NGT\Basic%20jQuery\prac%209\3.html', line 'Ln 16 Col 27', and other system information.

```
<!DOCTYPE html>
<html>
<head>
    <title></title>
    <script src="jquery-3.6.1.min.js"></script>
    <script>
        $(document).ready(function(){
            $("button").click(function(){
                $("p").css("color","red").slideUp(200).slideDown(200)
            })
        })
    </script>
</head>
<body>
    <p>Welcome Rohit</p>
    <button>Click me !</button>
</body>
</html>
```

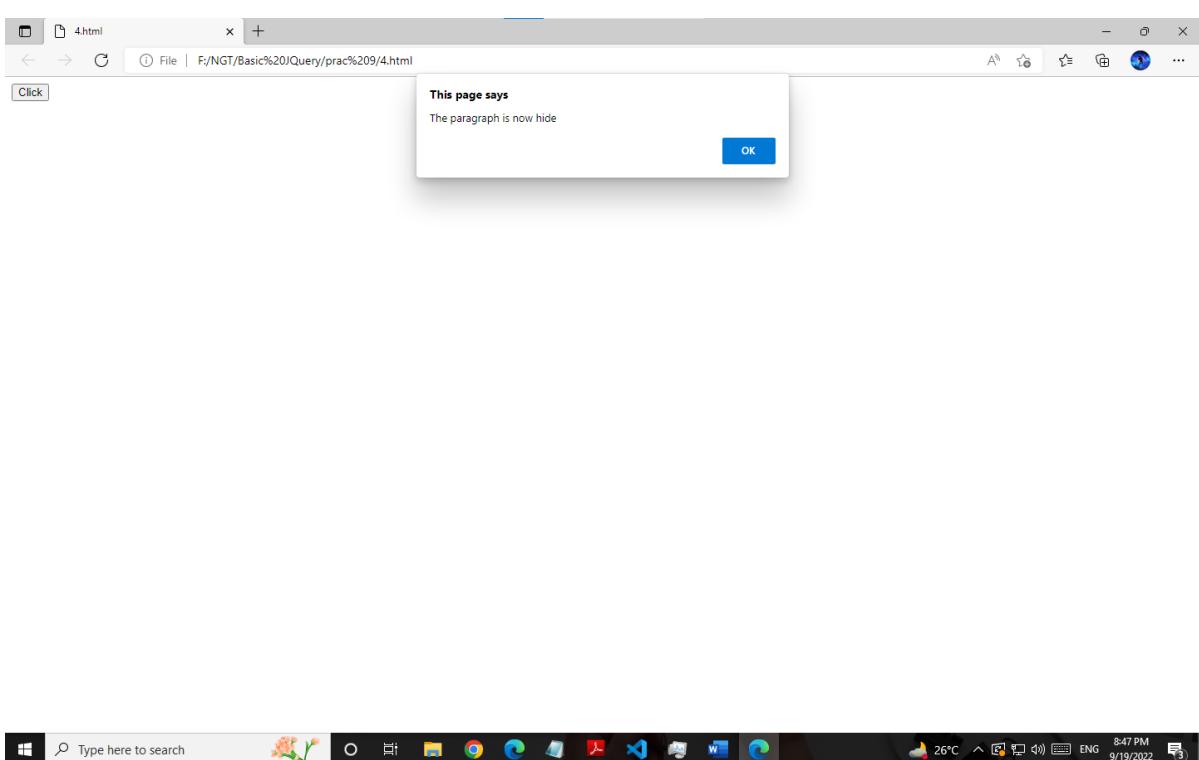
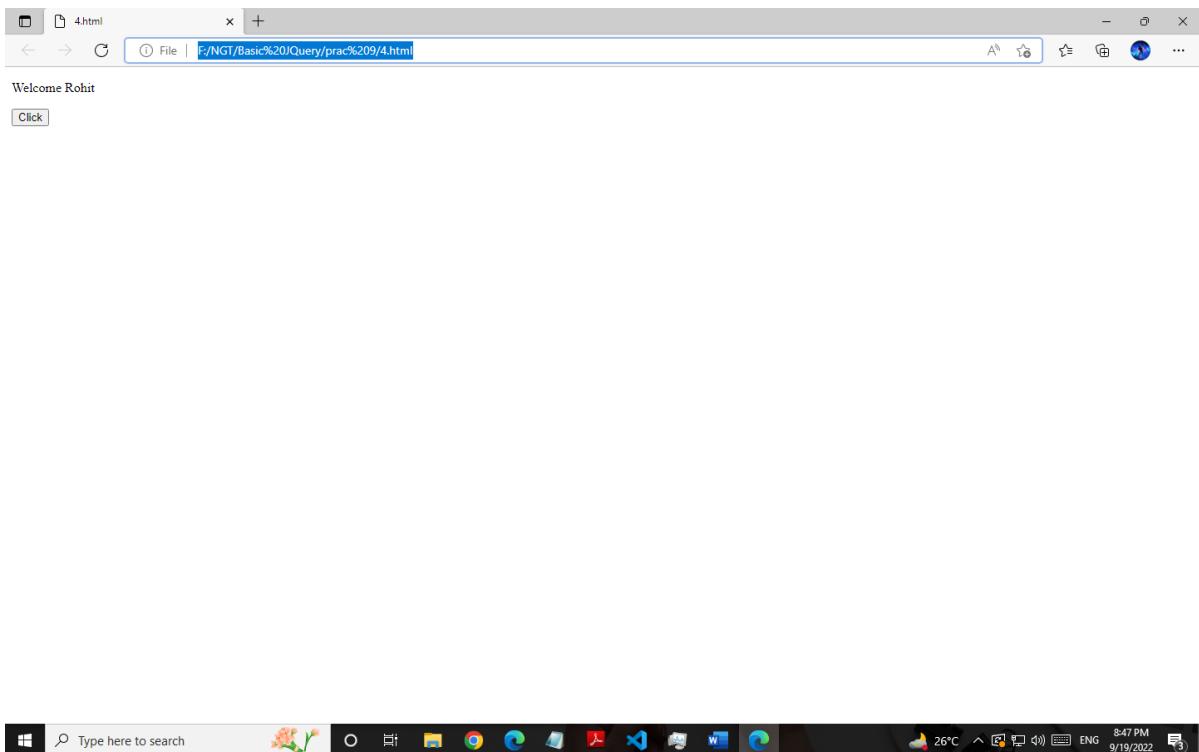




4. Write a jQuery effects method with callback function:

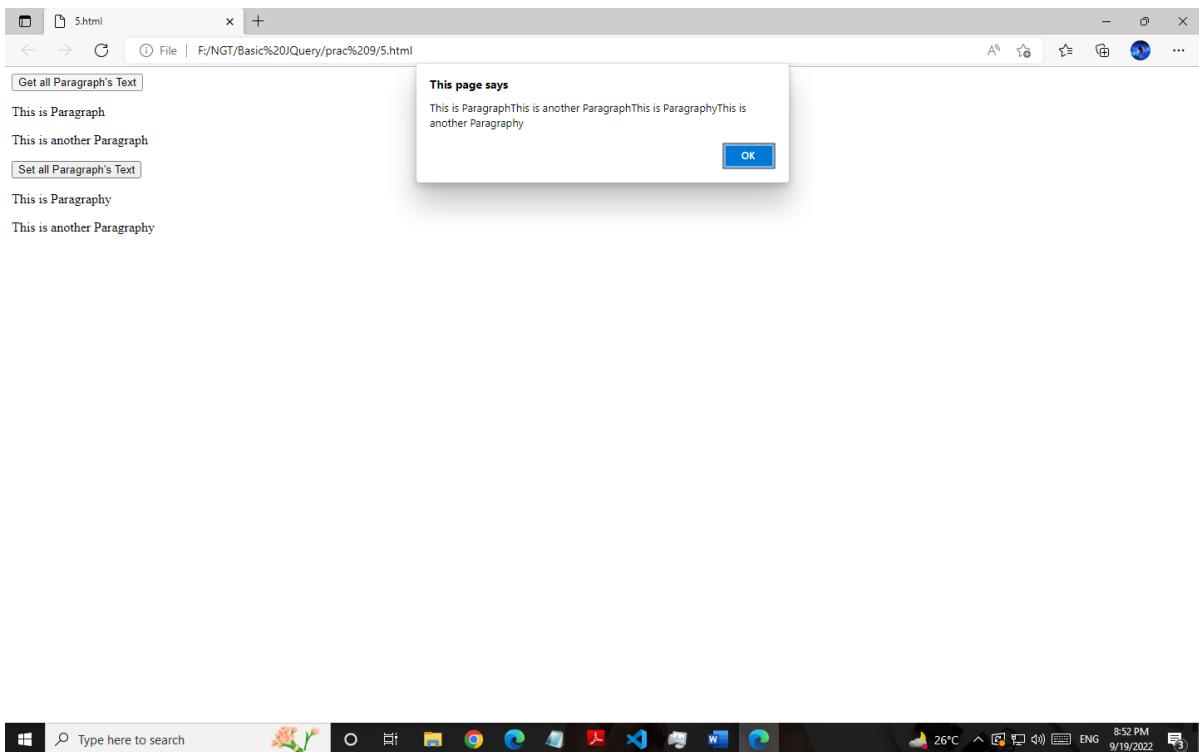
```
File Edit Selection View Go Run Terminal Help
EXT... 4.html
@idms-edgedevtools.
Microsoft Edge... 1M
Use the Microsoft Edge...
Microsoft Install
F > NGT > Basic JQuery > prac 9 > 4.html > html > head > script > ready() callback
1 <!DOCTYPE html>
2 <html>
3   <head>
4     <title></title>
5     <script src="jquery-3.6.1.min.js"></script>
6     <script>
7       $(document).ready(function(){
8         $("button").click(function(){
9           $("p").hide("slow",function(){
10             alert("The paragraph is now hide")
11           })
12         })
13       })
14     </script>
15   </head>
16   <body>
17     <p>Welcome Rohit</p>
18     <button>Click</button>
19   </body>
20 </html>
```

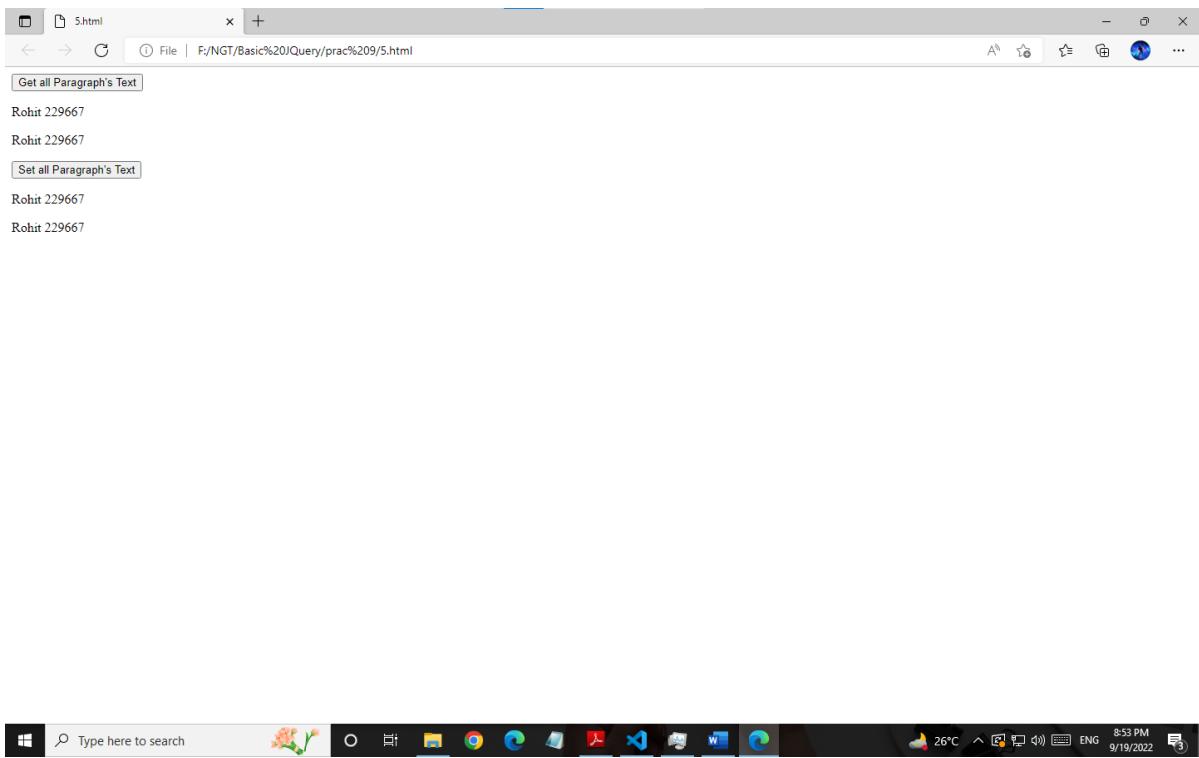
The screenshot shows a Visual Studio Code editor window with a dark theme. The file "4.html" is open, displaying a simple HTML document with a button and a paragraph. A jQuery script is attached to the document's ready event, which hides the paragraph when the button is clicked. The code uses the "slow" effect and includes a callback function to trigger an alert message. The status bar at the bottom of the screen shows system information like temperature, battery level, and date/time.



5. Write a jQuery to get and set text contents of the elements:

```
<!DOCTYPE html>
<html>
  <head>
    <title></title>
    <script src="jquery-3.6.1.min.js"></script>
    <script>
      $(document).ready(function(){
        $("#b1").click(function(){
          var str=$("#p").text();
          alert(str);
        });
        $("#b2").click(function(){
          $("p").text("Rohit 229667")
        })
      })
    </script>
  </head>
  <body>
    <button id = "b1">Get all Paragraph's Text</button>
    <p>This is Paragraph</p>
    <p>This is another Paragraph</p>
    <button id = "b2">Set all Paragraph's Text</button>
    <p>This is Paragraph</p>
    <p>This is another Paragraph</p>
  </body>
</html>
```





6. Write a jQuery to get and set HTML content of the elements:

A screenshot of Visual Studio Code showing the file "6.html". The code is as follows:

```
<!DOCTYPE html>
<html>
  <head>
    <title></title>
    <script src="jquery-3.6.1.min.js"></script>
    <script>
      $(document).ready(function(){
        $("#b1").click(function(){
          var str=$("#p").html()
          alert(str)
        });
        $("#b2").click(function(){
          $("body").html("<p> <b> Rohit 229667 </b> </p>")
        })
      })
    </script>
  </head>
  <body>
    <button id = "b1">Get Paragraph's HTML content</button>
    <p>The quick <b>brown fox</b>jumps over the lazy dog.</p>
    <button id = "b2">Write Message</button>
  </body>
</html>
```

The code uses jQuery to handle click events on buttons. The first button ("b1") retrieves the HTML content of the paragraph element and displays it in an alert. The second button ("b2") sets the entire body content to a new paragraph containing the text "Rohit 229667". The Visual Studio Code interface shows the code editor, a sidebar with file navigation, and a status bar at the bottom.

