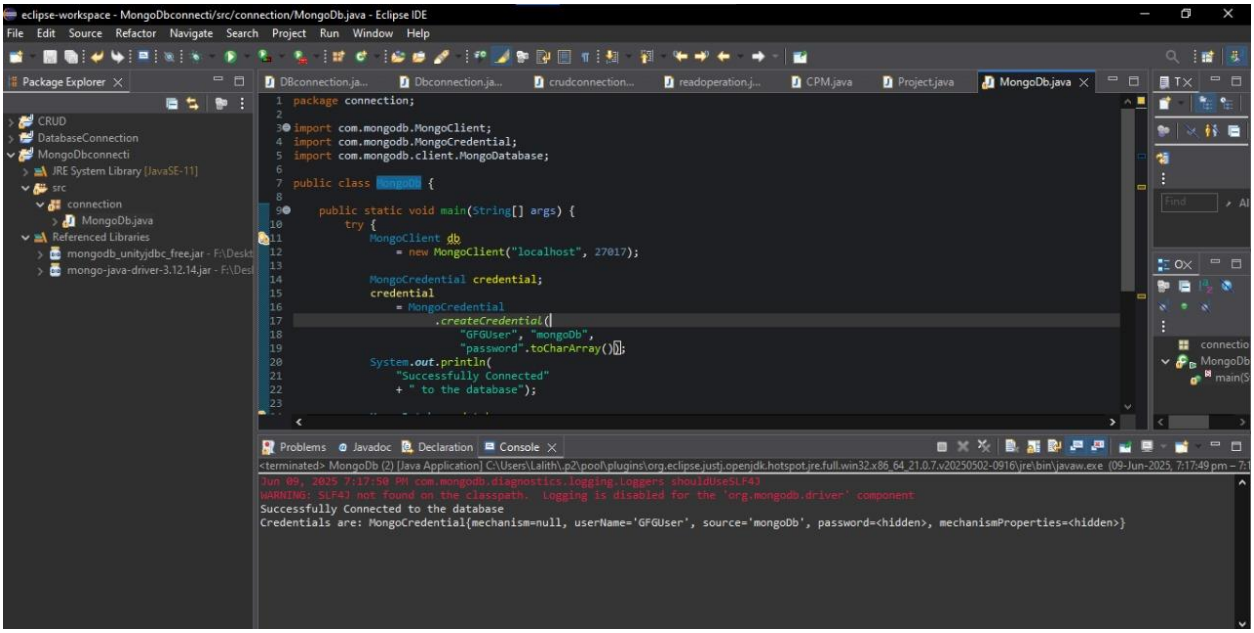
MONGODB

Ecllipse to mongodb connection:

```
package connection;

import com.mongodb.MongoClient;
import com.mongodb.MongoCredential;
import com.mongodb.client.MongoDatabase;

public class MongoDB {

    public static void main(String[] args) {
        try {
    MongoClient db
      = new MongoClient("localhost", 27017);

    MongoCredential credential;
    credential
      = MongoCredential
          .createCredential(
            "GFGUser", "mongoDb",
            "password".toCharArray());
    System.out.println(
      "Successfully Connected"
      + " to the database");

    MongoDatabase database
      = db.getDatabase("mongoDb");
    System.out.println("Credentials are: "
                + credential);
  }
```

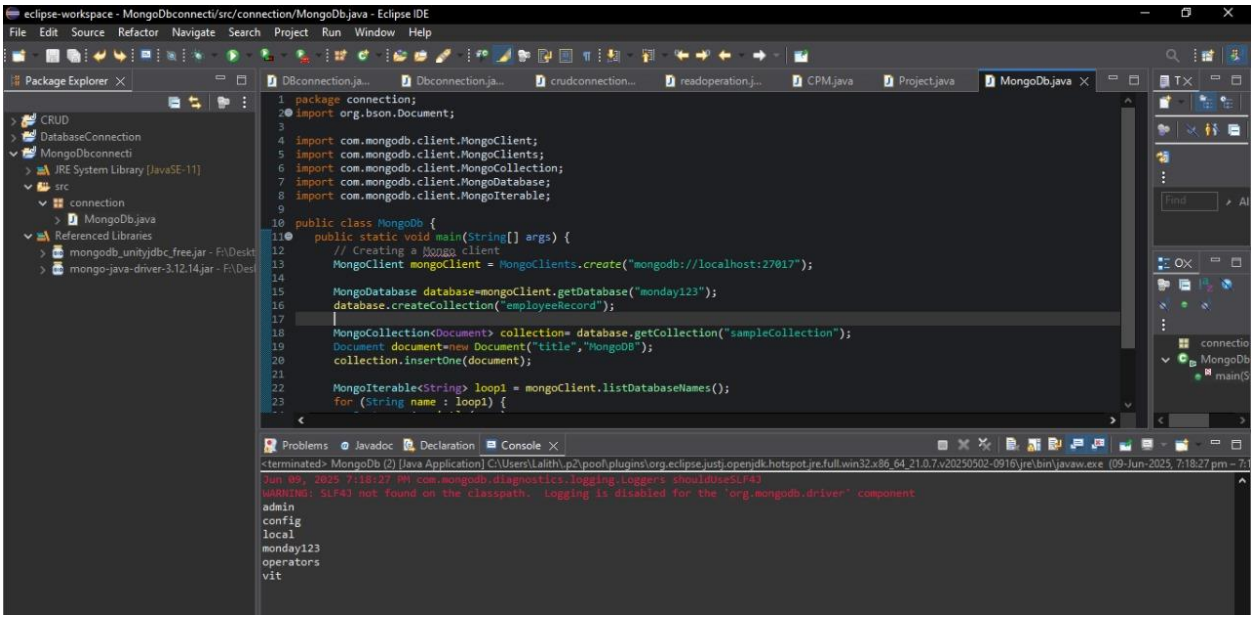
```
        catch (Exception e) {
            System.out.println(
                "Connection establishment failed");
            System.out.println(e);
        }

            }
}
```



Creating mongodb client:

```
package connection;
import org.bson.Document;

import com.mongodb.client.MongoClient;
import com.mongodb.client.MongoClients;
import com.mongodb.client.MongoCollection;
import com.mongodb.client.MongoDatabase;
import com.mongodb.client.MongoIterable;
```

```
public class MongoDb {
   public static void main(String[] args) {
      // Creating a Mongo client
      MongoClient mongoClient = MongoClients.create("mongodb://localhost:27017");

      MongoIterable<String> loop1 = mongoClient.listDatabaseNames();
      for (String name : loop1) {
         System.out.println(name);
      }

   }
}
```

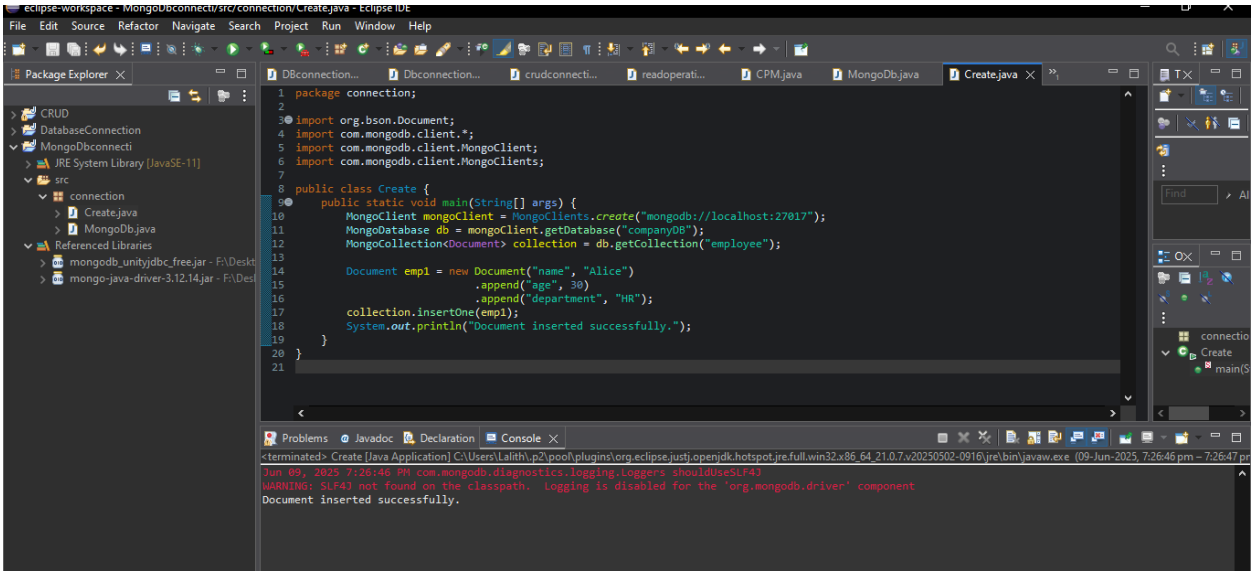


Create:

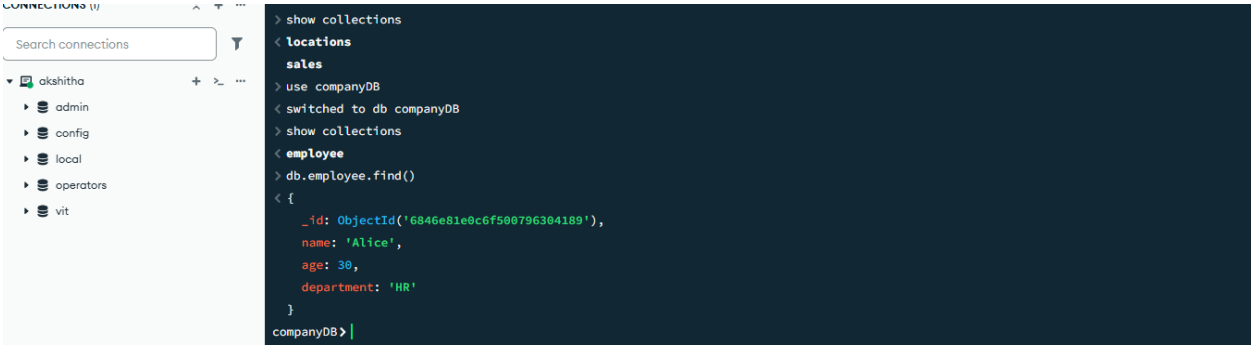
```
package connection;

import org.bson.Document;
import com.mongodb.client.*;
import com.mongodb.client.MongoClient;
import com.mongodb.client.MongoClients;
```

```
public class Create {
    public static void main(String[] args) {
        MongoClient mongoClient = MongoClients.create("mongodb://localhost:27017");
        MongoDatabase db = mongoClient.getDatabase("companyDB");
        MongoCollection<Document> collection = db.getCollection("employee");

        Document emp1 = new Document("name", "Alice")
                        .append("age", 30)
                        .append("department", "HR");
        collection.insertOne(emp1);
        System.out.println("Document inserted successfully.");
    }
}
```

Ecllipse:



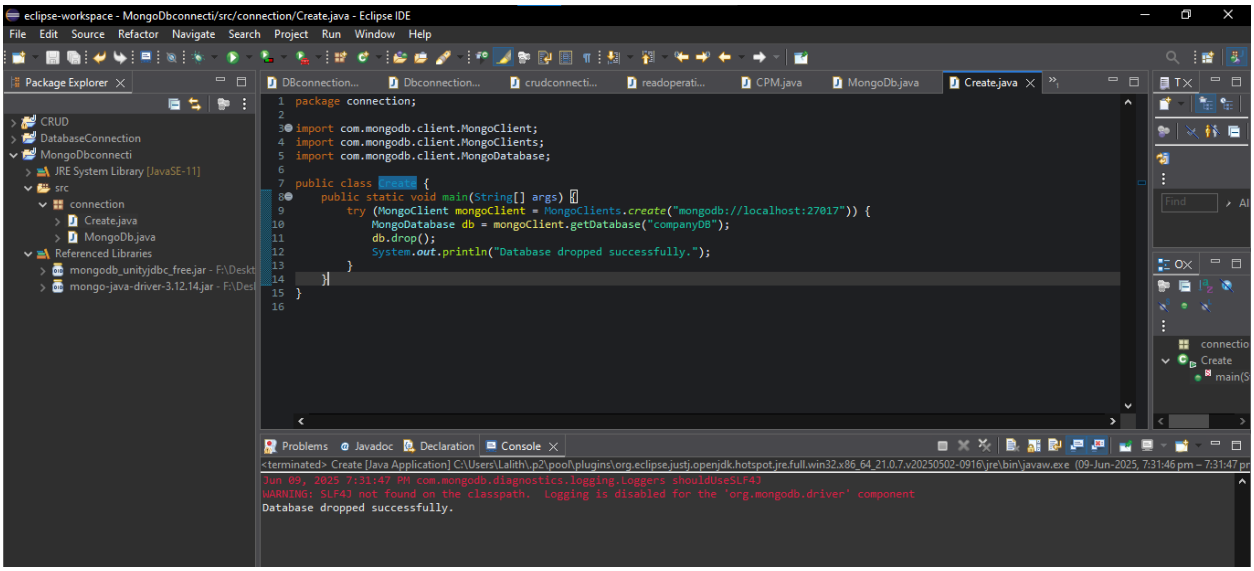
Mongodb:



DatabaseDropped:

```
package connection;


import com.mongodb.client.MongoClient;
import com.mongodb.client.MongoClients;
import com.mongodb.client.MongoDatabase;


public class Create {
    public static void main(String[] args) {
        try (MongoClient mongoClient = MongoClients.create("mongodb://localhost:27017")) {
            MongoDatabase db = mongoClient.getDatabase("companyDB");
            db.drop();
            System.out.println("Database dropped successfully.");
        }
    }
}
```

Ecllipse:



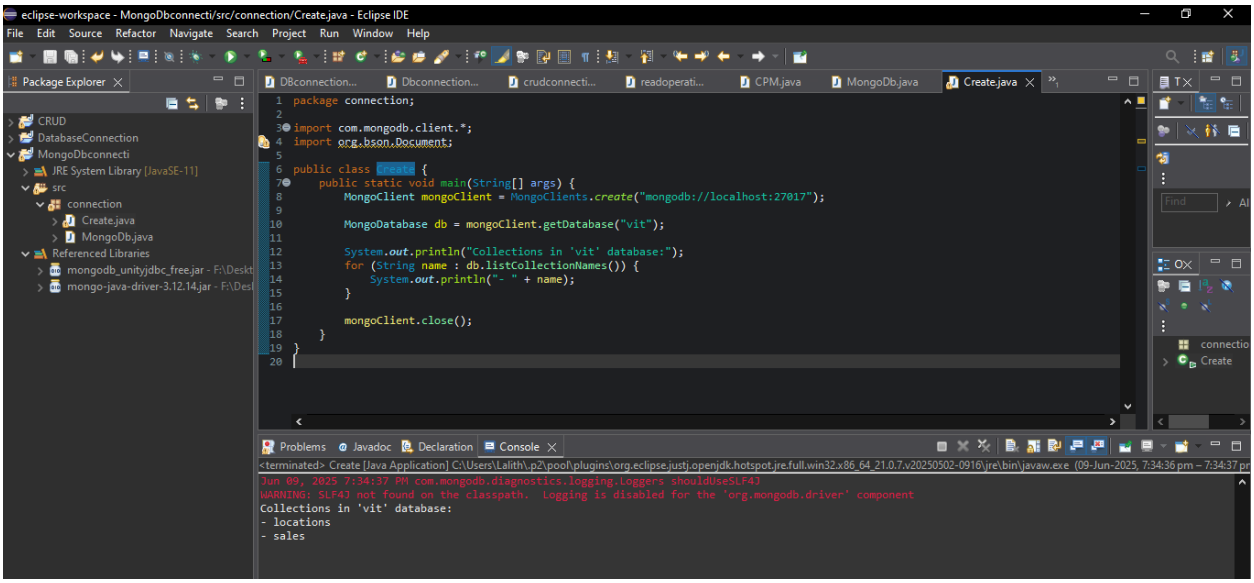
MONGODB:

Show collections:

```
package connection;


import com.mongodb.client.*;
import org.bson.Document;


public class Create {
    public static void main(String[] args) {
        MongoClient mongoClient = MongoClients.create("mongodb://localhost:27017");


        MongoDatabase db = mongoClient.getDatabase("vit");


        System.out.println("Collections in 'vit' database:");
        for (String name : db.listCollectionNames()) {
```

```
            System.out.println("- " + name);
        }

        mongoClient.close();
    }
}
```

Ecllipse:



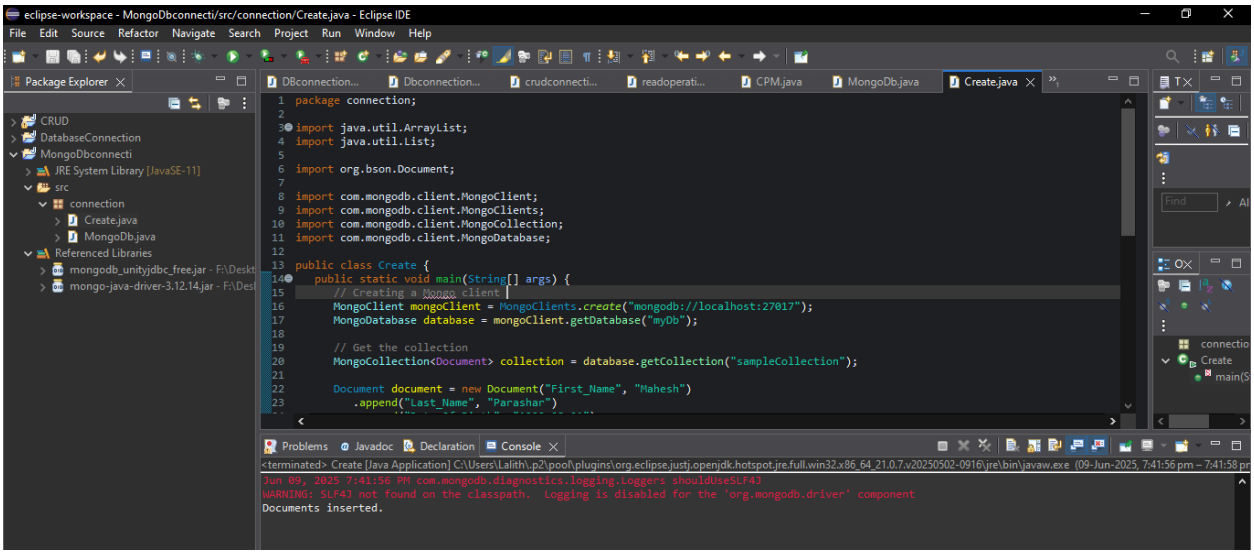
Inserting records:

```
package connection;

import java.util.ArrayList;
import java.util.List;

import org.bson.Document;

import com.mongodb.client.MongoClient;
import com.mongodb.client.MongoClients;
import com.mongodb.client.MongoCollection;
import com.mongodb.client.MongoDatabase;
```

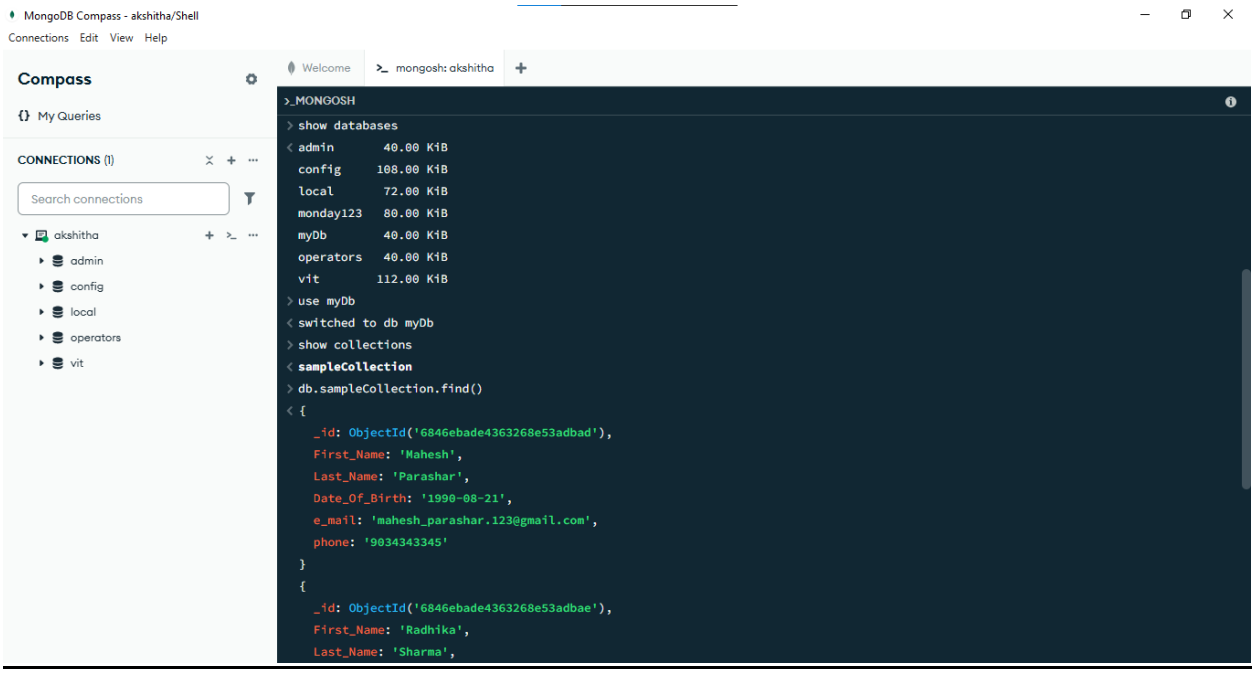
```
public class Create {
   public static void main(String[] args) {
      // Creating a Mongo client
      MongoClient mongoClient = MongoClients.create("mongodb://localhost:27017");
      MongoDatabase database = mongoClient.getDatabase("myDb");

      // Get the collection
      MongoCollection<Document> collection = database.getCollection("sampleCollection");

      Document document = new Document("First_Name", "Mahesh")
         .append("Last_Name", "Parashar")
         .append("Date_Of_Birth", "1990-08-21")
         .append("e_mail", "mahesh_parashar.123@gmail.com")
         .append("phone", "9034343345");

      collection.insertOne(document);
      List<Document> documents = new ArrayList<>();

      documents.add(new Document("First_Name", "Radhika")
         .append("Last_Name", "Sharma")
         .append("Date_Of_Birth", "1995-09-26")
         .append("e_mail", "radhika_sharma.123@gmail.com")
         .append("phone", "9000012345"));

      documents.add(new Document("First_Name", "Rachel")
         .append("Last_Name", "Christopher")
         .append("Date_Of_Birth", "1990-02-16")
         .append("e_mail", "Rachel_Christopher.123@gmail.com")
         .append("phone", "9000054321"));
```

```
    documents.add(new Document("First_Name", "Fathima")
        .append("Last_Name", "Sheik")
        .append("Date_Of_Birth", "1990-02-16")
        .append("e_mail", "Fathima_Sheik.123@gmail.com")
        .append("phone", "9000054321"));

    collection.insertMany(documents);

    System.out.println("Documents inserted.");
  }
}
```

Ecllipse:

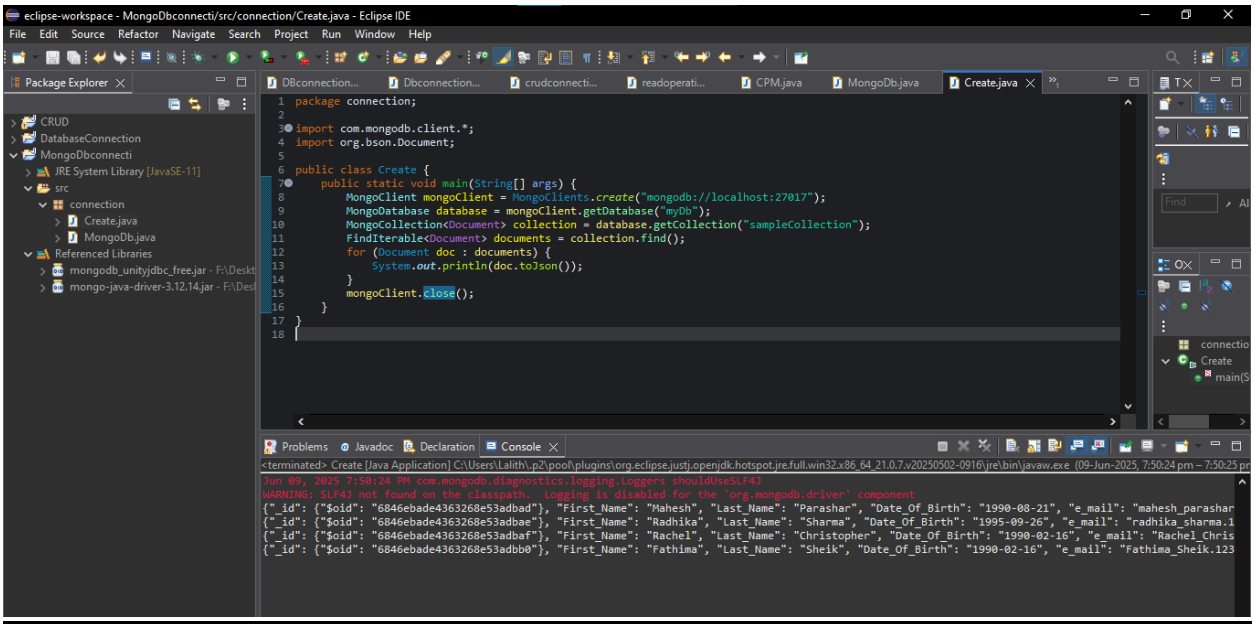


Mongodb:



Retrive the documents in ecllipse:

```
package connection;


import com.mongodb.client.*;
import org.bson.Document;


public class Create {
    public static void main(String[] args) {
        MongoClient mongoClient = MongoClients.create("mongodb://localhost:27017");
        MongoDatabase database = mongoClient.getDatabase("myDb");
        MongoCollection<Document> collection = database.getCollection("sampleCollection");
        FindIterable<Document> documents = collection.find();
        for (Document doc : documents) {
            System.out.println(doc.toJson());
        }
        mongoClient.close();
    }
}
```

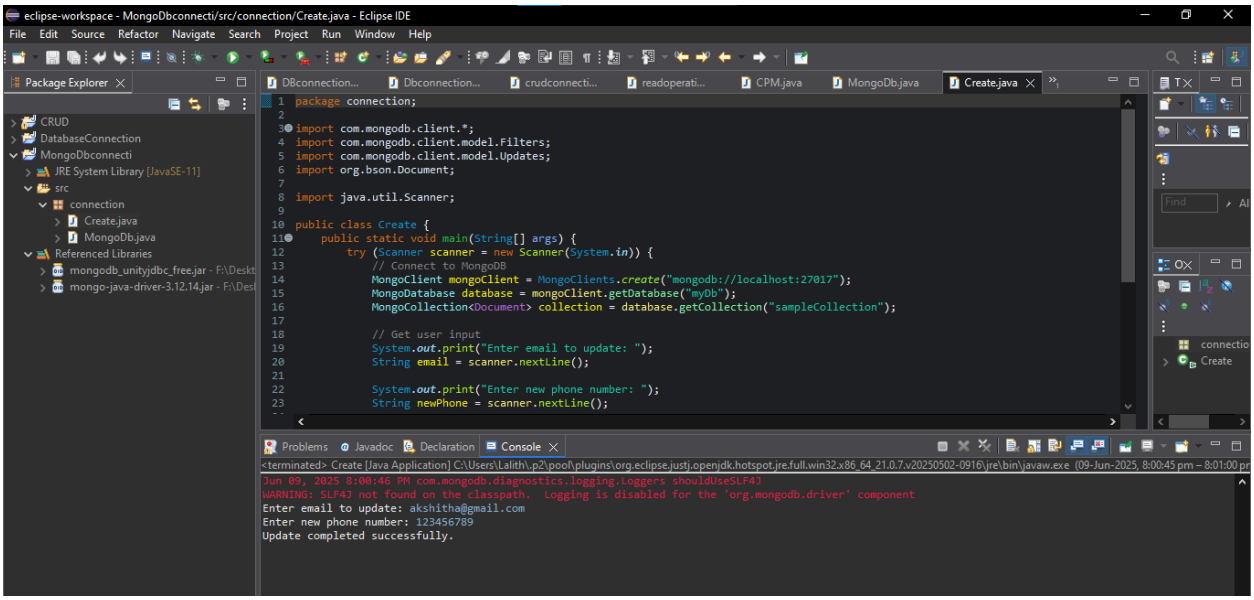


Update records:

```
package connection;


import com.mongodb.client.*;
import com.mongodb.client.model.Filters;
import com.mongodb.client.model.Updates;
import org.bson.Document;


import java.util.Scanner;


public class Create {
    public static void main(String[] args) {
        try (Scanner scanner = new Scanner(System.in)) {
            // Connect to MongoDB
            MongoClient mongoClient = MongoClients.create("mongodb://localhost:27017");
            MongoDatabase database = mongoClient.getDatabase("myDb");
            MongoCollection<Document> collection = database.getCollection("sampleCollection");
```

```
            // Get user input
            System.out.print("Enter email to update: ");
            String email = scanner.nextLine();

            System.out.print("Enter new phone number: ");
            String newPhone = scanner.nextLine();

            // Update document where e_mail matches input email
            collection.updateOne(
                Filters.eq("e_mail", email),
                Updates.set("phone", newPhone)
            );

            System.out.println("Update completed successfully.");

            mongoClient.close();
        } catch (Exception e) {
            e.printStackTrace();
        }
    }
}
```

Ecllipse:

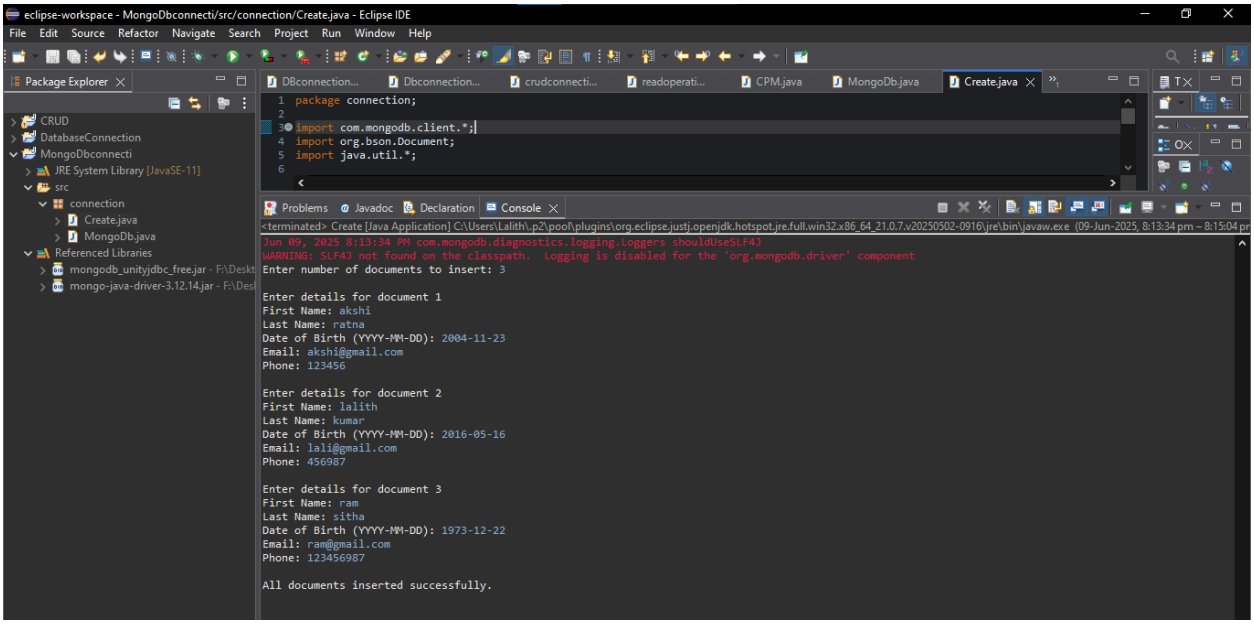


Udate many records:

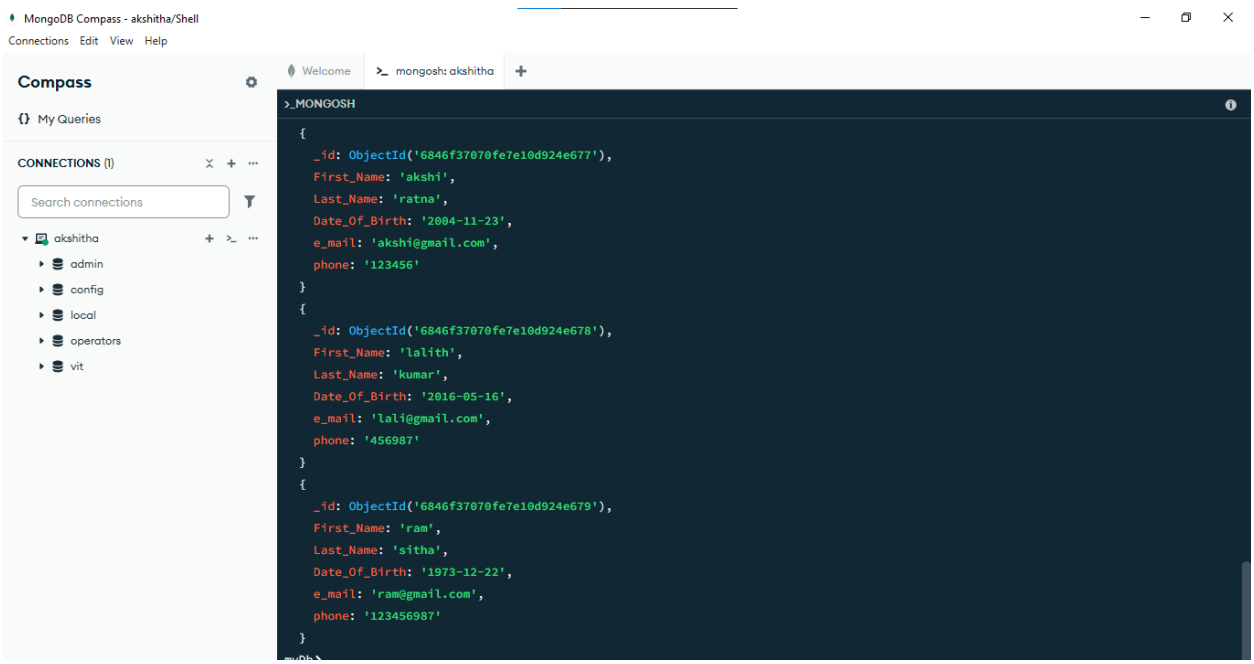
```
package connection;


import com.mongodb.client.*;

import org.bson.Document;

import java.util.*;


public class Create {

    public static void main(String[] args) {

        try (Scanner scanner = new Scanner(System.in)) {

            // Connect to MongoDB

            MongoClient mongoClient = MongoClients.create("mongodb://localhost:27017");

            MongoDatabase database = mongoClient.getDatabase("myDb");

            MongoCollection<Document> collection = database.getCollection("sampleCollection");


            System.out.print("Enter number of documents to insert: ");

            int count = scanner.nextInt();

            scanner.nextLine(); // consume newline
```

```
List<Document> documents = new ArrayList<>();

for (int i = 0; i < count; i++) {
    System.out.println("\nEnter details for document " + (i + 1));

    System.out.print("First Name: ");
    String firstName = scanner.nextLine();

    System.out.print("Last Name: ");
    String lastName = scanner.nextLine();

    System.out.print("Date of Birth (YYYY-MM-DD): ");
    String dob = scanner.nextLine();

    System.out.print("Email: ");
    String email = scanner.nextLine();

    System.out.print("Phone: ");
    String phone = scanner.nextLine();

    Document doc = new Document("First_Name", firstName)
            .append("Last_Name", lastName)
            .append("Date_Of_Birth", dob)
            .append("e_mail", email)
            .append("phone", phone);

    documents.add(doc);
}
```

```
            collection.insertMany(documents);
            System.out.println("\nAll documents inserted successfully.");

            mongoClient.close();
        } catch (Exception e) {
            e.printStackTrace();
        }
    }
}
```



Mongodb:


MongoDB Compass - akshitha/Shell
Connections Edit View Help
Compass
My Queries
CONNECTIONS (1)
Search connections
akshitha
admin
config
local
operators
vit
Welcome
mongosh: akshitha
>_MONGOSH
{
_id: ObjectId('6846f37070fe7e10d924e677'),
First_Name: 'akshi',
Last_Name: 'ratna',
Date_Of_Birth: '2004-11-23',
e_mail: 'akshi@gmail.com',
phone: '123456'
}
{
_id: ObjectId('6846f37070fe7e10d924e678'),
First_Name: 'lalith',
Last_Name: 'kumar',
Date_Of_Birth: '2016-05-16',
e_mail: 'lali@gmail.com',
phone: '456987'
}
{
_id: ObjectId('6846f37070fe7e10d924e679'),
First_Name: 'ram',
Last_Name: 'sitha',
Date_Of_Birth: '1973-12-22',
e_mail: 'ram@gmail.com',
phone: '123456987'
}