

Velagapudi Ramakrishna Siddhartha Engineering College

Kanuru, 520001



WEB PROGRAMMING AND DEVELOPMENT HOME ASSIGNMENT - 1

Code : 20IT6303

Batch Members :

208W1A12A99

208W1A12A0

208W1A12A1

Submitted To :

Dr . M . Ashok Kumar

Department Of IT

Question – 12

Use The Given attributes related to the company table and implement operations using Statement and Prepared Statement Interfaces and Perform CRUD Operations.

Program – 1 :

CRUD Operations using Statement

```
package javadb;
import java.io.*;
import java.sql.*;
public class Home_Assignment1
{
    public static void main(String[] args)
    {
        try
        {
            Class.forName("com.mysql.jdbc.Driver");
            Connection con =
            DriverManager.getConnection("jdbc:mysql://localhost/student","root","");
            Statement st = con.createStatement();

            //Create table
            String sqltable = "create table Company_Details(companyid int, companyname varchar(50),
            companynumber bigint, addressline varchar(500), city varchar(50), state varchar(50), postalcode
            int, country varchar(50))";
            //st.executeUpdate(sqltable);

            // Insert Records
            String sqlinsert = "insert into Company_Details values(2, 'EPAM', 456, 'Hyderbad India',
            'Telangana', 'India', 4987, 'India')";
            //st.executeUpdate(sqlinsert);

            // Update Query
            String sqlupdate = "update Company_Details set companyname='MicroSoft' where
            companyid = 1";
            //st.executeUpdate(sqlupdate);

            // Delete Query
            String sqldelete = "delete from Company_Details where companyid=1";
            st.executeUpdate(sqldelete);

            // Select Query
            ResultSet rs = st.executeQuery("select * from Company_Details");
```

```

while(rs.next())
{
    System.out.println(rs.getInt(1) + " | " + rs.getString(2) + " | " + rs.getInt(3) + " | " +
rs.getString(4) + " | " + rs.getString(5) + " | " + rs.getString(6) + " | " + rs.getInt(7) + " | " +
rs.getString(8));

}

con.close();

}
catch(Exception e)
{
    System.out.print("An UnKnown Error occured : \n\n" + e);
}
}
}

```

Outputs :

Create :

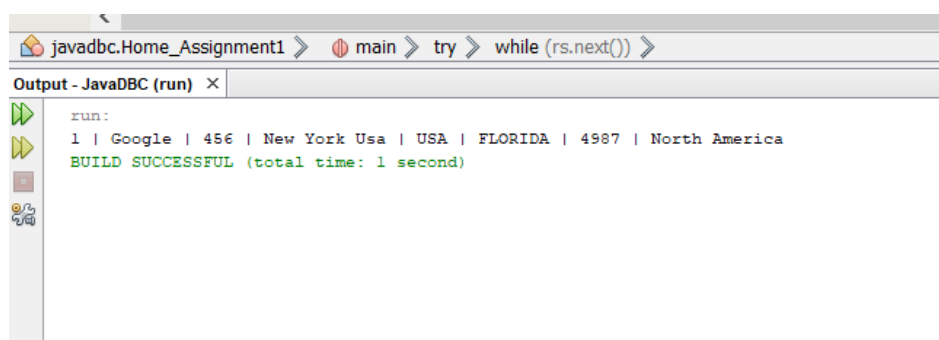
```

mysql> use student;
Database changed
mysql> desc company_details;
+-----+-----+-----+-----+-----+-----+
| Field          | Type          | Null | Key | Default | Extra |
+-----+-----+-----+-----+-----+-----+
| companyid      | int(11)       | YES  |     | NULL    |       |
| companyname    | varchar(50)   | YES  |     | NULL    |       |
| companynumber  | bigint(20)    | YES  |     | NULL    |       |
| addressline    | varchar(500)  | YES  |     | NULL    |       |
| city           | varchar(50)   | YES  |     | NULL    |       |
| state          | varchar(50)   | YES  |     | NULL    |       |
| postalcode     | int(11)       | YES  |     | NULL    |       |
| country        | varchar(50)   | YES  |     | NULL    |       |
+-----+-----+-----+-----+-----+-----+
8 rows in set (0.11 sec)

mysql>

```

Insert :

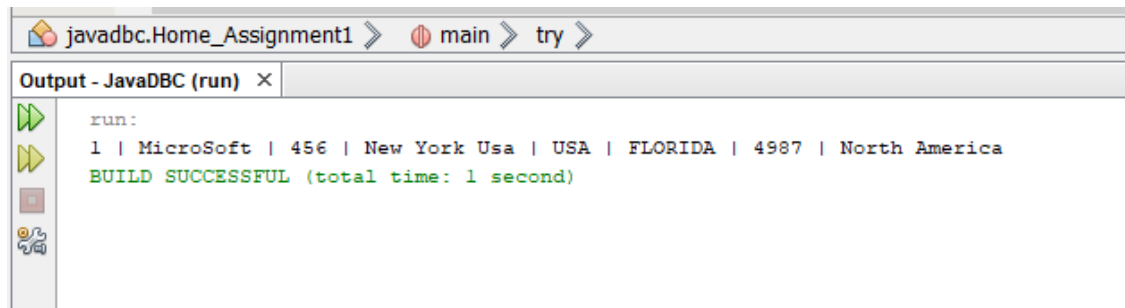


```

run:
1 | Google | 456 | New York Usa | USA | FLORIDA | 4987 | North America
BUILD SUCCESSFUL (total time: 1 second)

```

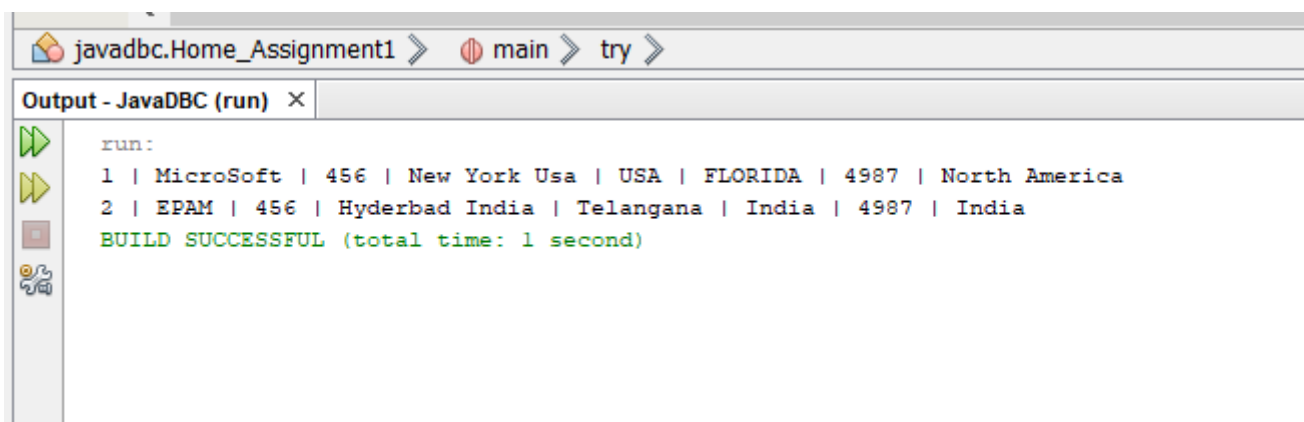
Update :



The screenshot shows an IDE window with the breadcrumb path `javadbc.Home_Assignment1 > main > try >`. Below it, the 'Output - JDBC (run)' console displays the following text:

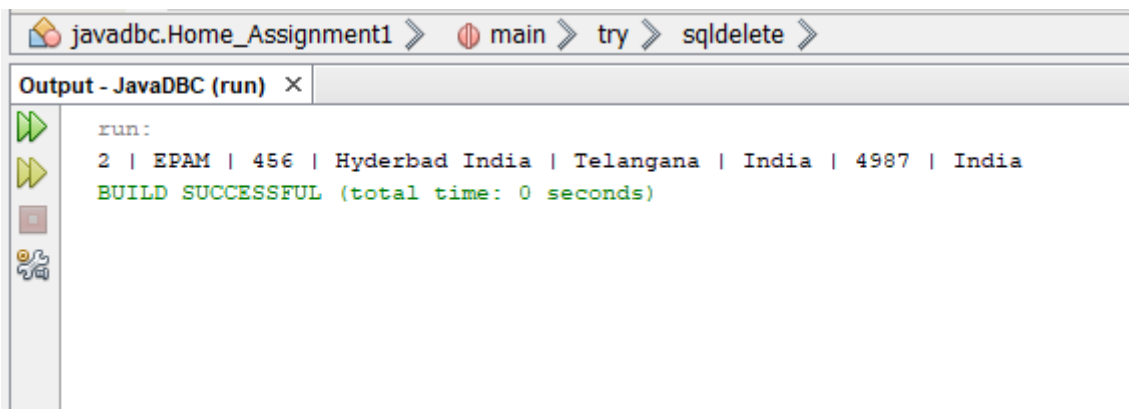
```
run:
1 | Microsoft | 456 | New York Usa | USA | FLORIDA | 4987 | North America
BUILD SUCCESSFUL (total time: 1 second)
```

Delete :



The screenshot shows an IDE window with the breadcrumb path `javadbc.Home_Assignment1 > main > try >`. Below it, the 'Output - JDBC (run)' console displays the following text:

```
run:
1 | Microsoft | 456 | New York Usa | USA | FLORIDA | 4987 | North America
2 | EPAM | 456 | Hyderabad India | Telangana | India | 4987 | India
BUILD SUCCESSFUL (total time: 1 second)
```



The screenshot shows an IDE window with the breadcrumb path `javadbc.Home_Assignment1 > main > try > sqldelete >`. Below it, the 'Output - JDBC (run)' console displays the following text:

```
run:
2 | EPAM | 456 | Hyderabad India | Telangana | India | 4987 | India
BUILD SUCCESSFUL (total time: 0 seconds)
```

Program – 2

CRUD Operation Using Prepared Statement

```
package javadb;
import java.io.*;
import java.sql.*;
public class Company_Home
{
    public static void main(String[] args)
    {
        try
        {
            Class.forName("com.mysql.jdbc.Driver");
            Connection con = DriverManager.getConnection("jdbc:mysql://localhost/college","root","");

            // Create table
            String sqltable = "create table Company_Details(companyid int, companyname varchar(50),
            companynumber bigint, addressline varchar(500), city varchar(50), state varchar(50), postalcode
            int, country varchar(50))";
            //      PreparedStatement psc = con.prepareStatement(sqltable);
            //      int t = psc.executeUpdate();
            //      System.out.println("Sucess Query : " + t);

            // Insert Query
            String sqlinsert = "insert into Company_Details values(2, 'Amazon', 456, 'Hyderbad India',
            'Telangana', 'India', 4987, 'India')";
            //      PreparedStatement ps = con.prepareStatement(sqlinsert);
            //      int t = ps.executeUpdate();
            //      System.out.println("Sucess Query : " + t);

            // Update Query
            String sqlupdate = "update Company_Details set companyname='MicroSoft' where
            companyid = 2";
            //      PreparedStatement ps = con.prepareStatement(sqlupdate);
            //      int t = ps.executeUpdate();
            //      System.out.println("Sucess Query : " + t);

            // Delete Query
            String sqldelete = "delete from Company_Details where companyid=2";
            PreparedStatement ps = con.prepareStatement(sqldelete);
            int t = ps.executeUpdate();
            System.out.println("Sucess Query : " + t);

            String sqlres = "select * from company_details";
            PreparedStatement ps1 = con.prepareStatement(sqlres);
            ResultSet rs = ps1.executeQuery();
            while(rs.next())
```

```

    {
        System.out.println(rs.getInt(1) + " | " + rs.getString(2) + " | " + rs.getInt(3) + " | " +
rs.getString(4) + " | " + rs.getString(5) + " | " + rs.getString(6) + " | " + rs.getInt(7) + " | " +
rs.getString(8));
    }
}
catch(Exception e)
{
    System.out.print("An UnKnown Error occured : \n\n" + e);
}
}
}

```

Outputs :

Create :

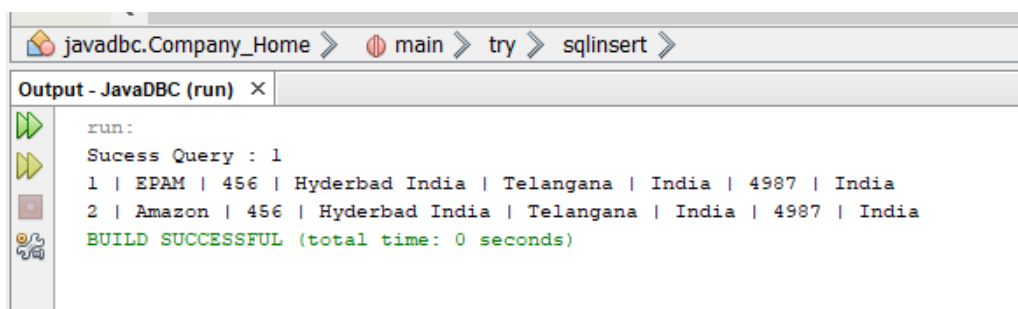
```

mysql> desc company_details;
+-----+-----+-----+-----+-----+-----+
| Field      | Type      | Null | Key | Default | Extra |
+-----+-----+-----+-----+-----+-----+
| companyid  | int(11)   | YES  |     | NULL    |       |
| companyname | varchar(50) | YES  |     | NULL    |       |
| companynumber | bigint(20) | YES  |     | NULL    |       |
| addressline | varchar(500) | YES  |     | NULL    |       |
| city       | varchar(50) | YES  |     | NULL    |       |
| state      | varchar(50) | YES  |     | NULL    |       |
| postalcode  | int(11)   | YES  |     | NULL    |       |
| country    | varchar(50) | YES  |     | NULL    |       |
+-----+-----+-----+-----+-----+-----+
8 rows in set (0.01 sec)

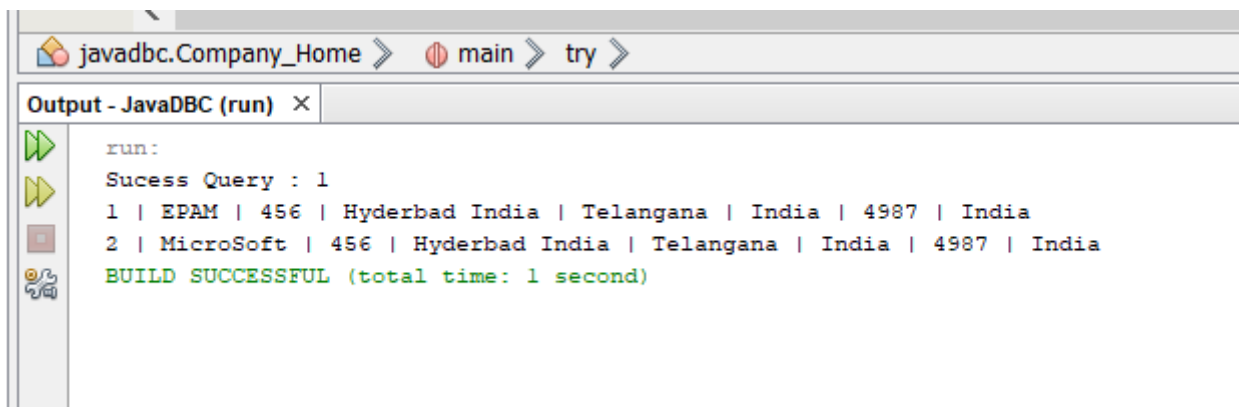
mysql>

```

Insert :



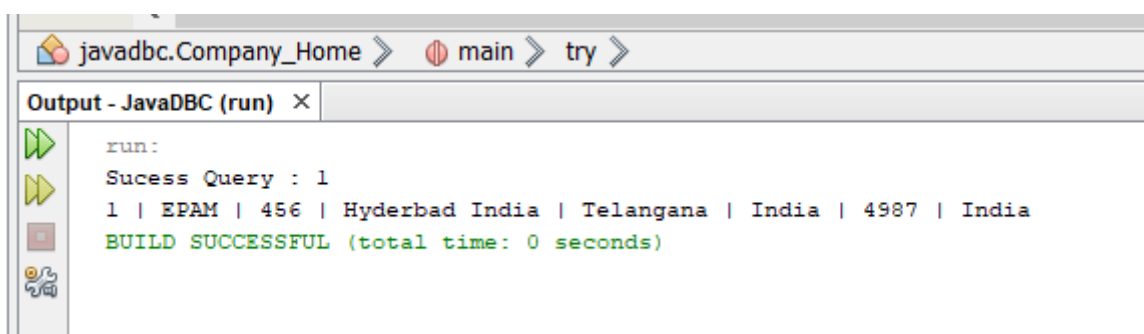
Update :



The screenshot shows an IDE window with the breadcrumb 'javadb.Company_Home > main > try'. Below it is a tab titled 'Output - JDBC (run)'. The output text is as follows:

```
run:
Sucess Query : 1
1 | EPAM | 456 | Hyderbad India | Telangana | India | 4987 | India
2 | MicroSoft | 456 | Hyderbad India | Telangana | India | 4987 | India
BUILD SUCCESSFUL (total time: 1 second)
```

Delete :

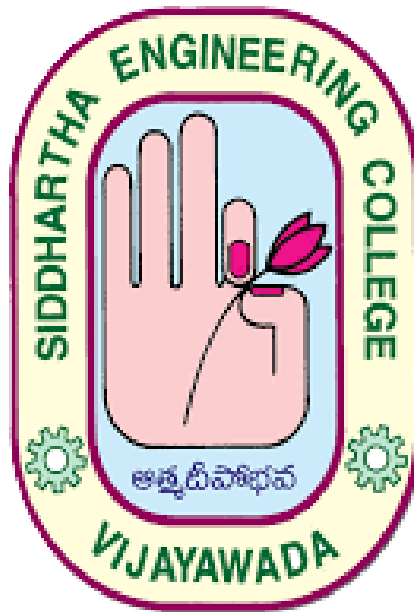


The screenshot shows an IDE window with the breadcrumb 'javadb.Company_Home > main > try'. Below it is a tab titled 'Output - JDBC (run)'. The output text is as follows:

```
run:
Sucess Query : 1
1 | EPAM | 456 | Hyderbad India | Telangana | India | 4987 | India
BUILD SUCCESSFUL (total time: 0 seconds)
```

Velagapudi Ramakrishna Siddhartha Engineering College

Kanuru, 520001



WEB PROGRAMMING AND DEVELOPMENT HOME ASSIGNMENT - 3

Code : 20IT6303

Batch Members :

208W1A12A99

208W1A12A0

208W1A12A1

Submitted To :

Dr . M . Ashok Kumar

Department Of IT

Question – 12

Create a Class Customer with four fields cno, cname, cmail and caddress.

Create a constructor, setters and getters.

Create a CustomerService class to insert the Customer details into the in-memory database and retrieve Customer information.

Test the application as a Spring Boot Application.

Program – 1 :

CDetails.java

```
package com.example.CustomerDetails;

public class CDetails
{
    private String cno, cname, cmail, caddress;

    public String getCno()
    {
        return cno;
    }

    public void setCno(String cno)
    {
        this.cno = cno;
    }

    public String getCname()
    {
        return cname;
    }

    public void setCname(String cname)
    {

```

```
this.cname = cname;

}

public String getCmail()
{
    return cmail;
}

public void setCmail(String cmail)
{
    this.cmail = cmail;
}

public String getAddress()
{
    return caddress;
}

public void setAddress(String caddress)
{
    this.caddress = caddress;
}

public CDetails(String cno, String cname, String cmail, String caddress)
{
    super();
    this.cno = cno;
    this.cname = cname;
    this.cmail = cmail;
    this.caddress = caddress;
}
}
```

Program – 2:

CService.java

```

package com.example.CustomerDetails;

import java.util.ArrayList;

import java.util.List;

import org.slf4j.Logger;

import org.slf4j.LoggerFactory;

import org.springframework.beans.factory.annotation.Autowired;

import org.springframework.jdbc.core.JdbcTemplate;

import org.springframework.stereotype.Service;

import javax.annotation.PostConstruct;

@Service

public class CService

{

    private static final Logger log = LoggerFactory.getLogger(CService.class);

    @Autowired

    JdbcTemplate jdbcTemplate;

    @PostConstruct

    public void postConstruct()

    {

        CDetails cd1 = new CDetails("1", "Prabhas", "actorprabhas@gmail.com", "Indian Film Industry, India");

        CDetails cd2 = new CDetails("2", "Ram Charan", "actorramcharan@gmail.com", "Indian Film Industry, India");

        List<CDetails> customers = new ArrayList<>();

        customers.add(cd1);

        customers.add(cd2);

        log.info("<----- Creating tables ----->");

        jdbcTemplate.execute("DROP TABLE Customer IF EXISTS");

        jdbcTemplate.execute("CREATE TABLE Customer(" + " cno varchar(255), cname varchar(255),

```

```

cmail varchar(255), address varchar(255))");

customers.forEach(i->jdbcTemplate.update("INSERT INTO Customer VALUES (?, ?, ?, ?)",
i.getCno(), i.getCname(), i.getCmail(), i.getCaddress()));

Log.info("<----- Records Saved ----->");

//retrieve saved records.

Log.info("<----- Retrieving records ----->");

customers = jdbcTemplate.query("select * from Customer", (rs, rowNum)-> new
CDetails(rs.getString("cno"), rs.getString("cname"), rs.getString("cmail"),
rs.getString("caddress")));

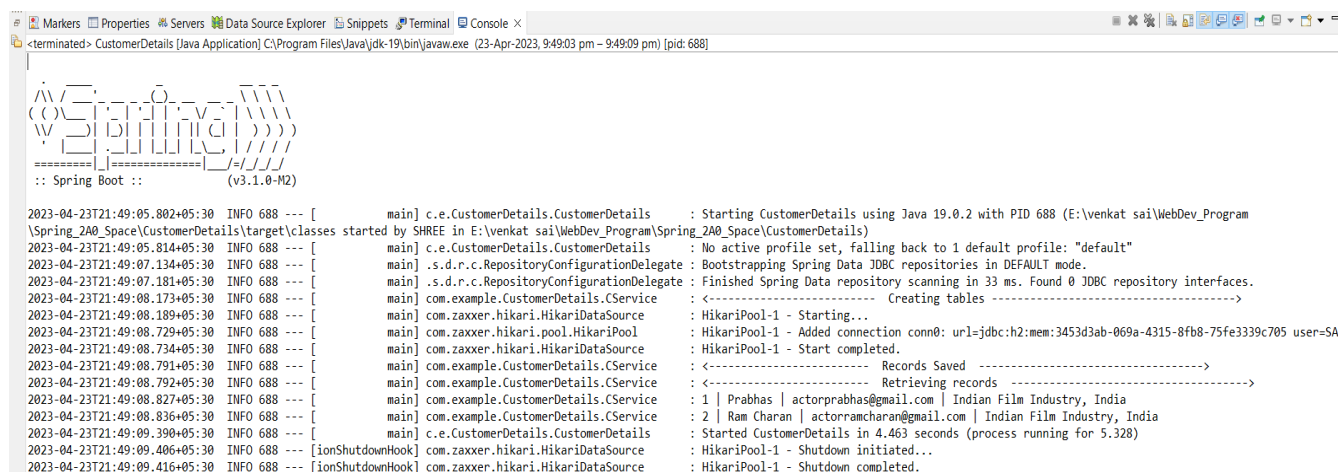
customers.forEach(i -> Log.info(i.getCno() + " | " + i.getCname() + " | " + i.getCmail()
+ " | " + i.getCaddress()));

}

}

```

Output :



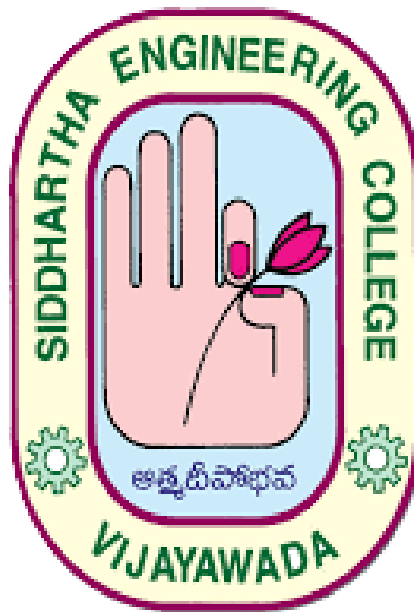
```

2023-04-23T21:49:05.802+05:30 INFO 688 --- [main] c.e.CustomerDetails.CustomerDetails : Starting CustomerDetails using Java 19.0.2 with PID 688 (E:\venkat sai\WebDev_Program\Spring_2A0_Space\CustomerDetails\target\classes started by SHREE in E:\venkat sai\WebDev_Program\Spring_2A0_Space\CustomerDetails)
2023-04-23T21:49:05.814+05:30 INFO 688 --- [main] c.e.CustomerDetails.CustomerDetails : No active profile set, falling back to 1 default profile: "default"
2023-04-23T21:49:07.134+05:30 INFO 688 --- [main] .s.d.r.c.RepositoryConfigurationDelegate : Bootstrapping Spring Data JDBC repositories in DEFAULT mode.
2023-04-23T21:49:07.181+05:30 INFO 688 --- [main] .s.d.r.c.RepositoryConfigurationDelegate : Finished Spring Data repository scanning in 33 ms. Found 0 JDBC repository interfaces.
2023-04-23T21:49:08.173+05:30 INFO 688 --- [main] com.example.CustomerDetails.CService : <----- Creating tables ----->
2023-04-23T21:49:08.189+05:30 INFO 688 --- [main] com.zaxxer.hikari.HikariDataSource : HikariPool-1 - Starting...
2023-04-23T21:49:08.729+05:30 INFO 688 --- [main] com.zaxxer.hikari.pool.HikariPool : HikariPool-1 - Added connection conn0: url=jdbc:h2:mem:3453d3ab-069a-4315-8fb8-75fe3339c705 user=SA
2023-04-23T21:49:08.734+05:30 INFO 688 --- [main] com.zaxxer.hikari.HikariDataSource : HikariPool-1 - Start completed.
2023-04-23T21:49:08.791+05:30 INFO 688 --- [main] com.example.CustomerDetails.CService : <----- Records Saved ----->
2023-04-23T21:49:08.792+05:30 INFO 688 --- [main] com.example.CustomerDetails.CService : <----- Retrieving records ----->
2023-04-23T21:49:08.827+05:30 INFO 688 --- [main] com.example.CustomerDetails.CService : 1 | Prabhas | actorprabhas@gmail.com | Indian Film Industry, India
2023-04-23T21:49:08.836+05:30 INFO 688 --- [main] com.example.CustomerDetails.CService : 2 | Ram Charan | actorramcharan@gmail.com | Indian Film Industry, India
2023-04-23T21:49:09.390+05:30 INFO 688 --- [main] c.e.CustomerDetails.CustomerDetails : Started CustomerDetails in 4.463 seconds (process running for 5.328)
2023-04-23T21:49:09.406+05:30 INFO 688 --- [ionShutdownHook] com.zaxxer.hikari.HikariDataSource : HikariPool-1 - Shutdown initiated...
2023-04-23T21:49:09.416+05:30 INFO 688 --- [ionShutdownHook] com.zaxxer.hikari.HikariDataSource : HikariPool-1 - Shutdown completed.

```

Velagapudi Ramakrishna Siddhartha Engineering College

Kanuru, 520001



WEB PROGRAMMING AND DEVELOPMENT HOME ASSIGNMENT - 4

Code : 20IT6303

Batch Members :

208W1A12A99

208W1A12A0

208W1A12A1

Submitted To :

Dr . M . Ashok Kumar

Department Of IT

Question – 12

Create a Class Customer with four fields cno, cname, cmail and caddress. Use parameterised constructor to set the values and write the getters to retrieve the data. Override the toString() method too. Create another CustomerController to return list of all objects as an array(the end point is /customers). Test the classes using Spring Boot.

Program – 1:

CDetails.java

```
package com.example.HomeAssignment4;

public class CDetails
{
    private String cno, cname, cmail, caddress;

    public String getCno()
    {
        return cno;
    }

    public void setCno(String cno)
    {
        this.cno = cno;
    }

    public String getCname()
    {
        return cname;
    }

    public void setCname(String cname)
    {
        this.cname = cname;
    }
}
```

```
}

public String getCmail()

{

return cmail;

}

public void setCmail(String cmail)

{

this.cmail = cmail;

}

public String getAddress()

{

return address;

}

public void setAddress(String address)

{

this.address = address;

}

public CDetails(String cno, String cname, String cmail, String address)

{

super();

this.cno = cno;

this.cname = cname;

this.cmail = cmail;

this.address = address;

}

}
```

Program – 2:

HomeApplication4.java

```
package com.example.HomeAssignment4;

import org.springframework.boot.SpringApplication;
import org.springframework.boot.autoconfigure.SpringBootApplication;

@SpringBootApplication
public class HomeAssignment4Application {

    public static void main(String[] args) {
        SpringApplication.run(HomeAssignment4Application.class, args);
    }

}
```

Program – 3:

CController.java

```
package com.example.HomeAssignment4;

import java.util.Arrays;
import java.util.List;
import org.springframework.web.bind.annotation.RequestMapping;
import org.springframework.web.bind.annotation.RestController;

@RestController

public class CController

{

    @RequestMapping("/customers")

    public List<CDetails> retrieveAllCourses()

    {

        return Arrays.asList(

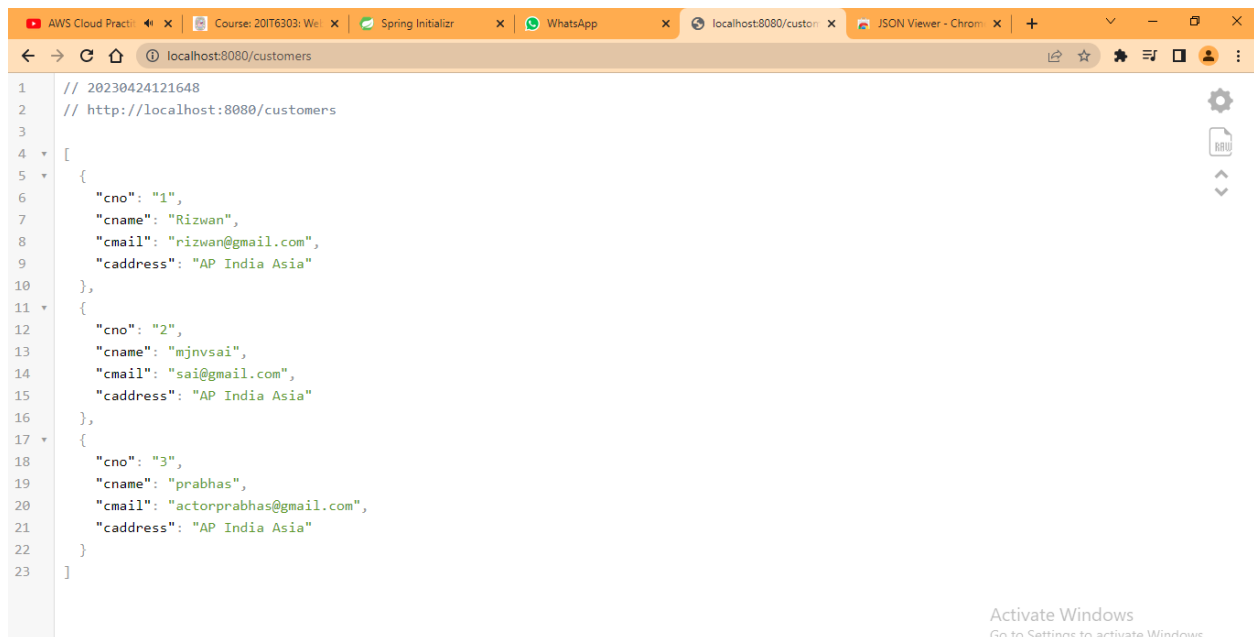
            new CDetails("1", "Rizwan", "rizwan@gmail.com", "AP India Asia"),


```



```
new CDetails("2", "mjnvsai", "sai@gmail.com", "AP India Asia"),  
  
new CDetails("3", "prabhas", "actorprabhas@gmail.com", "AP India Asia")  
  
);  
  
}  
  
}
```

Output :



```
1 // 20230424121648  
2 // http://localhost:8080/customers  
3  
4 [  
5   {  
6     "cno": "1",  
7     "cname": "Rizwan",  
8     "cmail": "rizwan@gmail.com",  
9     "caddress": "AP India Asia"  
10  },  
11   {  
12     "cno": "2",  
13     "cname": "mjnvsai",  
14     "cmail": "sai@gmail.com",  
15     "caddress": "AP India Asia"  
16  },  
17   {  
18     "cno": "3",  
19     "cname": "prabhas",  
20     "cmail": "actorprabhas@gmail.com",  
21     "caddress": "AP India Asia"  
22  }  
23 ]
```

Activate Windows
Go to Settings to activate Windows.