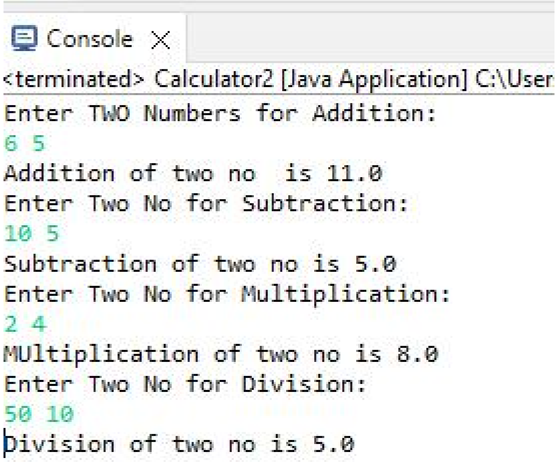
Q2. Write a Java program to Demonstrate a Generic Class To make Calculator.

```
package com.jatin2;

import java.io.*;
import java.util.*;

interface Calculator {
    abstract void Calculation();

    Scanner scanner = new Scanner(System.in);
}

class ADD implements  Calculator {
    public void Calculation() {
        System.out.println("Enter TWO Numbers for Addition:");
        float a = scanner.nextFloat();
        float b = scanner.nextFloat();

        float addition = (float)( a + b);
        System.out.println("Addition of two no  is " + addition);
    }
}

class Sub implements  Calculator {
    public void Calculation() {
        System.out.println("Enter Two No for Subtraction:");
        float a = scanner.nextFloat();
        float b = scanner.nextFloat();
        System.out.println("Subtraction of two no is " + (a - b));
    }
}

class MULT implements  Calculator {
    public void Calculation() {
        System.out.println("Enter Two No for Multiplication:");
        float a = scanner.nextFloat();
        float b = scanner.nextFloat();
        System.out.println("MUltiplication of two no is " + (a * b));
    }
}


class DIV implements  Calculator {
    public void Calculation() {
        System.out.println("Enter Two No for Division:");
        float a = scanner.nextFloat();
        float b = scanner.nextFloat();
        System.out.println("Division of two no is " + (a / b));
    }
}
public class Calculator2<T extends  Calculator> {
    T obj;

    Calculator2(T obj) {
```

```
                this.obj = obj;
        }

        void getObj() {
                this.obj.Calculation();
        }

        public static void main(String[] args) {
                Calculator2<ADD> c = new Calculator2<ADD>(new ADD());
                c.getObj();
                Calculator2<Sub> q = new Calculator2<Sub>(new Sub());
                q.getObj();
                Calculator2<MULT> s = new Calculator2<MULT>(new MULT());
                s.getObj();
                Calculator2<DIV> p = new Calculator2<DIV>(new DIV());
                p.getObj();

        }
}
```



```
<terminated> Calculator2 [Java Application] C:\User
Enter TWO Numbers for Addition:
6 5
Addition of two no  is 11.0
Enter Two No for Subtraction:
10 5
Subtraction of two no is 5.0
Enter Two No for Multiplication:
2 4
MUltiplication of two no is 8.0
Enter Two No for Division:
50 10
Division of two no is 5.0
```

Q.1 write program in Spring JDBC to demonstrate ResultExtractor Interface Faculty table

1.Pom.xml

```
<project xmlns="http://maven.apache.org/POM/4.0.0"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"

xsi:schemaLocation="http://maven.apache.org/POM/4.0.0
http://maven.apache.org/xsd/maven-4.0.0.xsd">

<modelVersion>4.0.0</modelVersion>

<groupId>com.jatin</groupId>

<artifactId>JDBC</artifactId>
```

```xml
<version>0.0.1-SNAPSHOT</version>
<packaging>jar</packaging>
<name>JDBC</name>
<url>http://maven.apache.org</url>
<properties>
<project.build.sourceEncoding>UTF-8</project.build.sourceEncoding>
</properties>
<dependencies>
<dependency>
<groupId>junit</groupId>
<artifactId>junit</artifactId>
<version>3.8.1</version>
<scope>test</scope>
</dependency>
<dependency>
<groupId>org.springframework</groupId>
<artifactId>spring-context</artifactId>
<version>6.0.4</version>
</dependency>
<!-- https://mvnrepository.com/artifact/org.springframework/spring-jdbc -->
<dependency>
<groupId>org.springframework</groupId>
<artifactId>spring-jdbc</artifactId>
<version>6.0.4</version>
</dependency>
```

```xml
<!-- https://mvnrepository.com/artifact/mysql/mysql-connector-java -->
<dependency>
<groupId>mysql</groupId>
<artifactId>mysql-connector-java</artifactId>
<version>8.0.32</version>
</dependency>
<dependency>
<groupId>junit</groupId>
<artifactId>junit</artifactId>
<version>4.11</version>
<scope>test</scope>
</dependency>
</dependencies>
</project>
```

2.Appliation.java

```java
package com.jatin.JDBC;
import java.util.List;
import org.springframework.context.ApplicationContext;
import org.springframework.context.support.ClassPathXmlApplicationContext;
public class Appliation {
public static void main(String[] args) {
System.out.println("---Example for ResultSetExtractor---");
ApplicationContext ctx = new ClassPathXmlApplicationContext("com/jatin/JDBC/Config.xml");
FacultyDAO dao = (FacultyDAO) ctx.getBean("fso");
List<Faculty> list = dao.getAllFaculties();
for (Faculty e : list)
System.out.println(e);
System.out.println("Executed By Jatin!!!");
}
}
```

3.Faculty.java

```
package com.jatin.JDBC;
public class Faculty {
private int id;
public int getId() {
return id;
}
public void setId(int id) {
this.id = id;
}
public String getName() {
return name;
}
public void setName(String name) {
this.name = name;
}
public float getSalary() {
return salary;
}
public void setSalary(float salary) {
```

```
this.salary = salary;
}
public Faculty(int id, String name, float salary) {
super();
this.id = id;
this.name = name;
this.salary = salary;
}
public Faculty() {
super();
}
private String name;
private float salary;
public String toString() {
return id + " " + name + " " + salary;
}
}
```

4. FacyultyDAO.java

```
package com.jatin.JDBC;
import java.sql.ResultSet;
import java.sql.SQLException;
import java.util.ArrayList;
import java.util.List;
import org.springframework.dao.DataAccessException;
import org.springframework.jdbc.core.JdbcTemplate;
import org.springframework.jdbc.core.ResultSetExtractor;

public class FacultyDAO {

private JdbcTemplate jdbcTemplate;

public void setJdbcTemplate(JdbcTemplate jdbcTemplate) {

this.jdbcTemplate = jdbcTemplate;

}

public List<Faculty> getAllFaculties() {

return jdbcTemplate.query("select * from faculty", new
ResultSetExtractor<List<Faculty>>() {

public List<Faculty> extractData(ResultSet rs) throws SQLException,
DataAccessException {

List<Faculty> list = new ArrayList<Faculty>();

while (rs.next()) {

Faculty e = new Faculty();

e.setId(rs.getInt(1));

e.setName(rs.getString(2));

e.setSalary(rs.getInt(3));

list.add(e);

}

return list;

}

});

}

}
```

5. Config.xml

```
<?xml version="1.0" encoding="UTF-8"?>
```

```xml
<beans

xmlns="http://www.springframework.org/schema/beans"

xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"

xmlns:p="http://www.springframework.org/schema/p"

xsi:schemaLocation="http://www.springframework.org/schema/beans

http://www.springframework.org/schema/beans/spring-beans-3.0.xsd">

<bean id="ds"
class="org.springframework.jdbc.datasource.DriverManagerDataSource">

<property name="driverClassName" value="com.jatin.JDBC.jdbc.Driver"/>

<property name="url" value="jdbc:mysql://localhost:3306/springjdbc" />

<property name="username" value="root"/>

<property name="password" value="root"/>

</bean>

<bean id="jdbcTemplate" class="org.springframework.jdbc.core.JdbcTemplate">

<property name="dataSource" ref="ds"></property>

</bean>

<bean id="fso" class="com.spring.jdbc.pract3.EmployeeDao">

<property name="jdbcTemplate" ref="jdbcTemplate"></property>

</bean>

</beans>
```

```sql
SQL:-
CREATE DATABASE springjdbc;
USE springjdbc;
CREATE TABLE sports(Id INT PRIMARY KEY NOT NULL AUTO_INCREMENT,Name
VARCHAR(20),salary VARCHAR(20));
insert into employee values (101,'Jatin',75000);
insert into employee values (102,'Pranav',70000);
insert into employee values (103,Rajesh',65000);
```

```
Mysql> select *from faculty;

Id    Name     salary

101  Jatin   75000
102  Pranav  70000
103  Rajesh  65000
```