

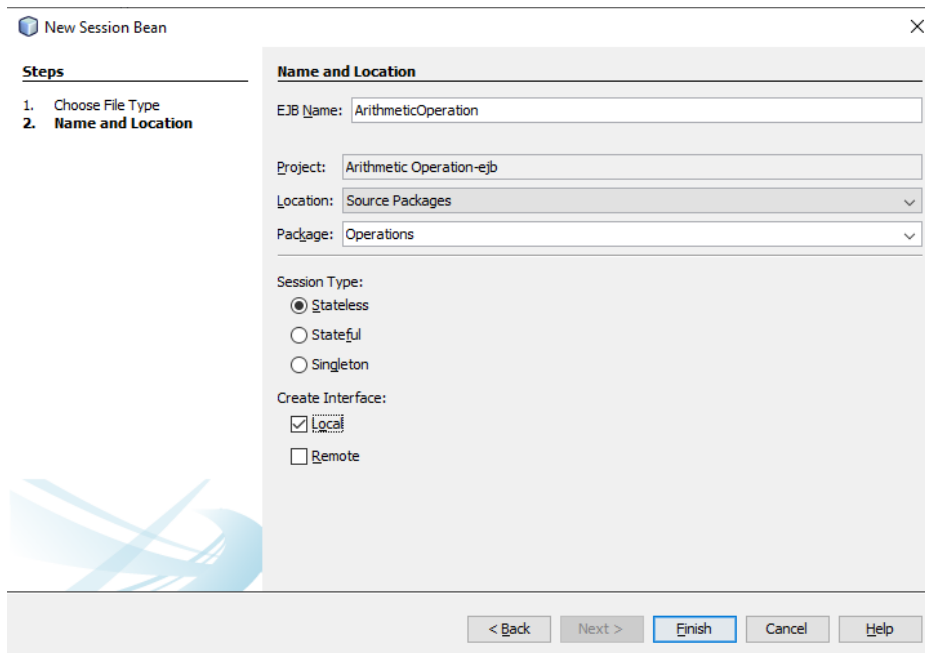
Practical No.: 6.1

Aim: Sample program for basic arithmetic operations implemented in session bean.

File → New Project → JavaEE → Enterprise Application → Project Name: Arithmetic Operation → Finish

New Session Bean for Arithmetic Operation

Right click on EJB Module Project (Arithmetic Operation-ejb) → New → Session Bean



The 'New Session Bean' dialog box is shown with the 'Name and Location' tab selected. The 'EJB Name' is 'ArithmeticOperation'. The 'Project' is 'Arithmetic Operation-ejb'. The 'Location' is 'Source Packages' and the 'Package' is 'Operations'. Under 'Session Type', 'Stateless' is selected. Under 'Create Interface', 'Local' is checked and 'Remote' is unchecked. The 'Finish' button is highlighted.

Steps

1. Choose File Type
2. **Name and Location**

Name and Location

EJB Name: ArithmeticOperation

Project: Arithmetic Operation-ejb

Location: Source Packages

Package: Operations

Session Type:

☒ Stateless

☐ Stateful

☐ Singleton

Create Interface:

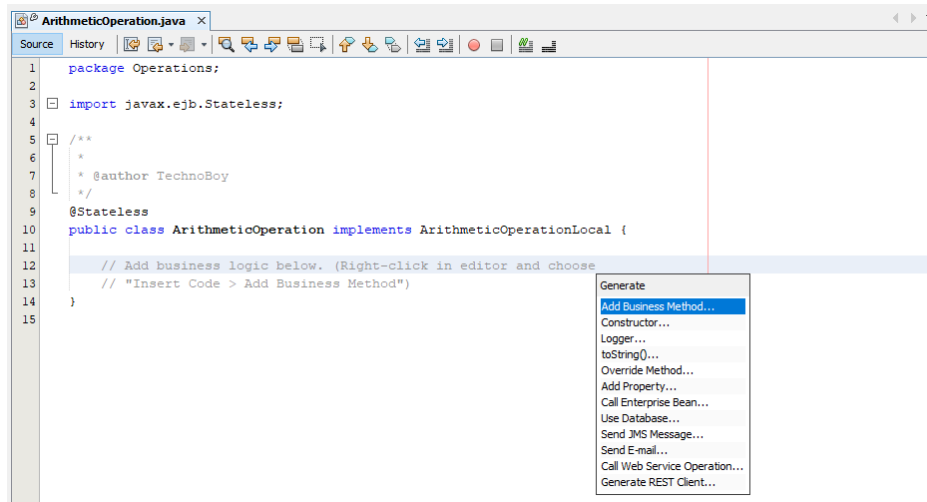
☒ Local

☐ Remote

< Back Next > **Finish** Cancel Help

ArithmeticOperation.java

Right click on coding area and click on Insert Code → Add Business Method



The code editor shows the 'ArithmeticOperation.java' file. The code is as follows:

```
1 package Operations;
2
3 import javax.ejb.Stateless;
4
5 /**
6  *
7  * @author TechnoBoy
8  */
9 @Stateless
10 public class ArithmeticOperation implements ArithmeticOperationLocal {
11
12     // Add business logic below. (Right-click in editor and choose
13     // "Insert Code > Add Business Method")
14
15 }
```

A context menu is open over the code, showing the 'Generate' option selected, with 'Add Business Method...' highlighted.

Generate

- Add Business Method...
- Constructor...
- Logger...
- toString()
- Override Method...
- Add Property...
- Call Enterprise Bean...
- Use Database...
- Send JMS Message...
- Send E-mail...
- Call Web Service Operation...
- Generate REST Client...

AddBusinessMethod

Dialog: Add Business Method...

Name: Add

Return Type: int [Browse...]

Parameters Exceptions

Name	Type	Final
x	int	<input type="checkbox"/>
y	int	<input type="checkbox"/>

[Add] [Remove] [Up] [Down]

Use in Interface: ☒ Local ☐ Remote ☐ Both

[OK] [Cancel]

Repeat the same process for Subtraction, Multiplication and Division

ArithmeticOperation.java

```
package Operation;
```

```
import javax.ejb.Stateless;
```

```
/**
```

```
 *
```

```
 * @author TechnoBoy
```

```
 */
```

```
@Stateless
```

```
public class ArithmeticOperation implements ArithmeticOperationLocal {
```

```
    @Override
```

```
    public int Add(int x, int y) {
```

```
        return x+y;
```

```
    }
```

```

@Override
public int Minus(int x, int y) {
    return x-y;
}

@Override
public int Multiply(int x, int y) {
    return x*y;
}

@Override
public int Divide(int x, int y) {
    return x/y;
}
}

```

Right click on EJB Web Project (**Arithmetic Operation-war**) → New → Servlet → Class Name :
calculation

Calculation.java

```

package ArithmeticOperations;

import Operation.ArithmeticOperationLocal;
import java.io.IOException;
import java.io.PrintWriter;
import javax.ejb.EJB;
import javax.servlet.ServletException;
import javax.servlet.http.HttpServlet;
import javax.servlet.http.HttpServletRequest;
import javax.servlet.http.HttpServletResponse;

/**
 *
 * @author TechnoBoy
 */
public class calculation extends HttpServlet {

    @EJB
    private ArithmeticOperationLocal arithmeticOperation;
    int add,sub,mul,div;
    protected void processRequest(HttpServletRequest request, HttpServletResponse response)
        throws ServletException, IOException {
        response.setContentType("text/html;charset=UTF-8");
    }
}

```

```

try (PrintWriter out = response.getWriter()) {
    out.println("<!DOCTYPE html>");
    out.println("<html>");
    out.println("<head>");
    out.println("<title>Servlet calculation</title>");
    out.println("</head>");
    out.println("<body>");
    out.println("Addition : "+add+"<br>");
    out.println("Substraction : "+sub+"<br>");
    out.println("Multiplication : "+mul+"<br>");
    out.println("Division : "+div+"<br>");
    out.println("</body>");
    out.println("</html>");
}
}
@Override
protected void doGet(HttpServletRequest request, HttpServletResponse response)
    throws ServletException, IOException {
    processRequest(request, response);
}
@Override
protected void doPost(HttpServletRequest request, HttpServletResponse response)
    throws ServletException, IOException {
    PrintWriter out = response.getWriter();
    int a=Integer.parseInt(request.getParameter("a"));
    int b=Integer.parseInt(request.getParameter("b"));
    add=arithmeticOperation.Add(a, b);
    sub=arithmeticOperation.Minus(a, b);
    mul=arithmeticOperation.Multiply(a, b);
    div=arithmeticOperation.Divide(a, b);
    processRequest(request, response);
}
@Override
public String getServletInfo() {
    return "Short description";
}
}

```

Right click on EJB Web Project (**Arithmetic Operation-war**) → New → JSP → File Name: index

Index.jsp

```

<%--
    Document : index
    Created on : 19 Oct, 2019, 1:02:04 AM
    Author : TechnoBoy
--%>

<%@page contentType="text/html" pageEncoding="UTF-8"%>
<!DOCTYPE html>
<html>
    <head>

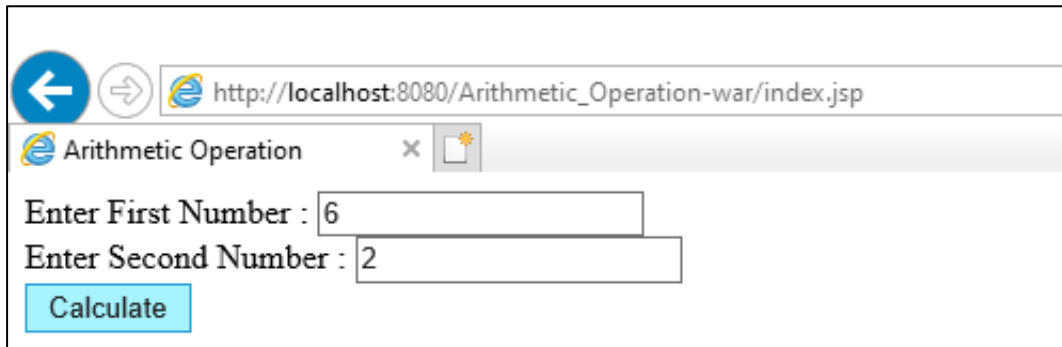
```

```
<meta http-equiv="Content-Type" content="text/html; charset=UTF-8">
<title>Arithmetic Operation</title>
</head>
<body>
  <form action="calculation" method="post">
    Enter First Number : <input type="number" name="a" placeholder="Enter 1st Number"><br>
    Enter Second Number : <input type="number" name="b" placeholder="Enter 2nd
Number"><br>
    <input type="submit" value="Calculate">
  </form>
</body>
</html>
```

Output

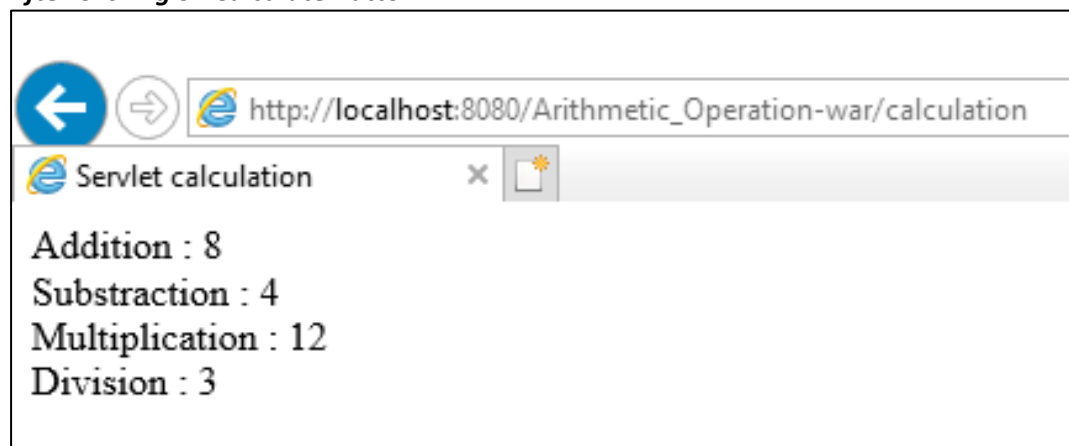
Right click on Enterprise Application Project (Arithemtic Operation) and Click on **Clean and Build** then click on **Deploy** and lastly click on **Run**

Before Clicking on Calculate Button



The screenshot shows a web browser window with the address bar displaying `http://localhost:8080/Arithmetic_Operation-war/index.jsp`. The browser tab is titled "Arithmetic Operation". The page content includes two input fields: "Enter First Number : 6" and "Enter Second Number : 2". Below these fields is a blue button labeled "Calculate".

After Clicking on Calculate Button



The screenshot shows the same web browser window, but the address bar now displays `http://localhost:8080/Arithmetic_Operation-war/calculation`. The browser tab is titled "Servlet calculation". The page content displays the results of the calculations: "Addition : 8", "Substraction : 4", "Multiplication : 12", and "Division : 3".