```
public class MatrixOperations {
    public Integer rows { get; set; }
    public Integer cols { get; set; }

    public List<List<Integer>> matrix1 { get; set; }
    public List<List<Integer>> matrix2 { get; set; }
    public List<List<Integer>> resultMatrix { get; set; }

    // Constructor
    public MatrixOperations() {
        matrix1 = new List<List<Integer>>();
        matrix2 = new List<List<Integer>>();
        resultMatrix = new List<List<Integer>>();
    }

    // Create empty matrices with default values (0s)
    public void createMatrices() {
        matrix1 = new List<List<Integer>>();
        matrix2 = new List<List<Integer>>();
        for (Integer i = 0; i < rows; i++) {
            List<Integer> row1 = new List<Integer>();
            List<Integer> row2 = new List<Integer>();
            for (Integer j = 0; j < cols; j++) {
                row1.add(0);
                row2.add(0);
            }
            matrix1.add(row1);
            matrix2.add(row2);
```

```
        }
        resultMatrix.clear();
    }

    // Add two matrices
    public void addMatrices() {
        resultMatrix = new List<List<Integer>>();
        if (matrix1.size() != matrix2.size() || matrix1[0].size() != matrix2[0].size()) {
            ApexPages.addMessage(new ApexPages.Message(ApexPages.Severity.ERROR,
'Matrix sizes do not match.'));
            return;
        }

        for (Integer i = 0; i < rows; i++) {
            List<Integer> row = new List<Integer>();
            for (Integer j = 0; j < cols; j++) {
                row.add(matrix1[i][j] + matrix2[i][j]);
            }
            resultMatrix.add(row);
        }
    }
}
```

```
<apex:page controller="MatrixOperations">
  <h2>Matrix Addition</h2>
  <apex:form>
    <apex:pageMessages />

    <!-- Input rows and columns -->
    <apex:pageBlock title="Enter Matrix Dimensions">
      <apex:pageBlockSection columns="2">
        <apex:inputText value="{!rows}" label="Number of Rows"/>
        <apex:inputText value="{!cols}" label="Number of Columns"/>
        <apex:commandButton value="Create Matrices" action="{!createMatrices}"
rerender="matrixInputs,resultPanel"/>
      </apex:pageBlockSection>
    </apex:pageBlock>

    <!-- Matrix input -->
    <apex:pageBlock id="matrixInputs" title="Enter Matrix Values"
rendered="{!AND(NOT(ISNULL(rows)), NOT(ISNULL(cols)))}">
      <apex:pageBlockSection title="Matrix 1">
        <apex:repeat value="{!matrix1}" var="row" id="m1">
          <apex:repeat value="{!row}" var="element">
            <apex:inputText
value="{!matrix1[matrix1.indexOf(row)][row.indexOf(element)]}" style="width:40px;
margin:2px;"/>
          </apex:repeat>
          <br/>
        </apex:repeat>
      </apex:pageBlockSection>
```

```
        <apex:pageBlockSection title="Matrix 2">
          <apex:repeat value="{!matrix2}" var="row" id="m2">
            <apex:repeat value="{!row}" var="element">
              <apex:inputText
value="{!matrix2[matrix2.indexOf(row)][row.indexOf(element)]}" style="width:40px;
margin:2px;"/>
            </apex:repeat>
            <br/>
          </apex:repeat>
        </apex:pageBlockSection>

        <apex:pageBlockSection>
          <apex:commandButton value="Add Matrices" action="{!addMatrices}"
rerender="resultPanel, matrixInputs"/>
        </apex:pageBlockSection>
      </apex:pageBlock>

      <!-- Result Matrix -->
      <apex:outputPanel id="resultPanel">
        <apex:pageBlock title="Matrix Addition Result"
rendered="{!NOT(ISNULL(resultMatrix))}">
          <apex:pageBlockSection>
            <apex:repeat value="{!resultMatrix}" var="row">
              <apex:repeat value="{!row}" var="element">
                <apex:outputText value="{!element}" style="display:inline-block;
width:40px; text-align:center; margin:2px;" />
              </apex:repeat>
              <br/>
            </apex:repeat>
```

```
            </apex:pageBlockSection>
        </apex:pageBlock>
    </apex:outputPanel>
  </apex:form>
</apex:page>

// Create an instance of the controller
MatrixOperations matrixOps = new MatrixOperations();

// Set matrix dimensions
matrixOps.rows = 3;
matrixOps.cols = 3;

// Initialize sample matrices (3x3)
matrixOps.matrix1 = new List<List<Integer>>{
  new List<Integer>{1, 2, 3},
  new List<Integer>{4, 5, 6},
  new List<Integer>{7, 8, 9}
};

matrixOps.matrix2 = new List<List<Integer>>{
  new List<Integer>{9, 8, 7},
  new List<Integer>{6, 5, 4},
  new List<Integer>{3, 2, 1}
};

// Perform matrix addition
matrixOps.addMatrices();
```

```
// Debug output
System.debug('Result Matrix:');
for (List<Integer> row : matrixOps.resultMatrix) {
    System.debug(row);
}
```