Program to demonstrate Built-in functions of String Class **Code:**

```
import java.util.Scanner;
class StringFunctions
{
    public static void main(String args[])
    {
        String str1 = "    Hello";
        String str2 = "World";
        String str3 = "From Java";
        String str4 = "In Java";
        String str5 = str3.concat(" ").concat(str4);
        System.out.println("Returns 0 if str1 == \"Hello\"" + str1.contentEquals("Hello"))
        System.out.println("str1 == \"Hello\"" + str1.equals("Hello"));        // equals()
        System.out.println( "str1 == \"Hello\" After trimming:"str1.trim().equals("Hello"))
        System.out.println("str2.compareToIgnoreCase(\"world\"):" + str2.compareToIgnoreCas
        System.out.println("" + str2.toLowerCase().equalsIgnoreCase("World"));// toLowerCa
        System.out.println( + str3.toUpperCase()); // toUpperCase()
        System.out.println( + str5.replaceFirst("Java", "Command Prompt")); // replaceFirs
        System.out.println("Replace " + str5.replaceAll("Java", "Command Prompt")); // rep
        System.out.println( + str5.replaceAll("Java", "Command Prompt").contains("Java"));
        System.out.println("str5 ends with \"Java\"?" + str5.endsWith("Java")); // endsWit
        StringBuilder s = new StringBuilder(str3); // For using contentEquals which takes
        System.out.println(str3.contentEquals(s)); // contentEquals()
        System.out.println("".isEmpty()); // isEmpty()
        System.out.println(str4.replace('I', 'O')); // replace()
        System.out.println(str5.length()); // length()
        System.out.println(str5.charAt(3)); // charAt()
        System.out.println(str5.substring(str5.indexOf("Java"))); // substring()
        char arr[] = str1.toCharArray(); // toCharArray()
        for(int i = 0; i < arr.length; i++)
        {
            if(arr[i] == ' ')
            {
                arr[i] = '_';
            }
        }
        System.out.println(arr);
    }
}
```

Output:

Matrix Class:

```
// matrix/Matrix.java
package matrix;
```

```java
import java.util.Scanner;
public class Matrix
{
    int arr[][];
    int rows, columns;
    public Matrix(int rows, int columns)
    {
        arr = new int[rows][columns];
        this.rows = rows;
        this.columns = columns;
    }
    public Matrix()
    {
        arr = new int[2][2];
        rows = 2;
        columns = 2;
    }
    public int elementAt(int row, int column)
    {
        return arr[row][column];
    }
    public void setElement(int row, int column, int data)
    {
        arr[row][column] = data;
    }
    public void setMatrix()
    {
        Scanner sc = new Scanner(System.in);
        for(int i = 0; i < rows; i++)
        {
            for(int j = 0; j < columns; j++)
            {
                System.out.print("mat[" + i +"]" + "[" + j + "]: ");
                this.setElement(i, j, sc.nextInt());
            }
        }
    }
    public String toString()
    {
        StringBuilder str = new StringBuilder();
        for(int i=0; i < rows; i++)
        {
            for(int j = 0; j < columns; j++)
            {
                str.append(this.elementAt(i, j));
                str.append(' ');
            }
```

```
            str.append('\n');
        }
        return str.toString();
    }

    public Matrix transpose()
    {
        Matrix matTranspose = new Matrix(rows, columns);
        for(int i = 0; i < rows; i++)
        {
            for(int j = 0; j < columns; j++)
            {
                matTranspose.setElement(i, j, this.elementAt(j, i));
            }
        }
        return matTranspose;
    }

    public boolean equals(Matrix mat)
    {
        if( rows != mat.columns || columns != mat.columns)
        {
            System.out.print("Cannot Compare these matrices");
        }
        for(int i = 0; i < rows; i++)
        {
            for(int j = 0; j < columns; j++)
            {
                if(this.elementAt(i, j) != mat.elementAt(i, j))
                {
                    return false;
                }
            }
        }
        return true;
    }

    public int getColumns()
    {
        return columns;
    }

    public int getRows()
    {
        return rows;
    }
}
```

To check if the entered matrix is symmetric or not **Code:**

```
import java.util.Scanner;
import matrix.Matrix;
class Symmetric
{
    static boolean isSymmetric(Matrix mat)
    {
        return mat.equals(mat.transpose());
    }
    public static void main(String args[])
    {
        Scanner sc = new Scanner(System.in);
        System.out.print("Enter the order of the Matrix: ");
        int order = sc.nextInt();
        Matrix mat2 = new Matrix(order, order);
        for(int i = 0; i < order; i++)
        {
            for(int j = 0; j < order; j++)
            {
                System.out.print("mat[" + i +"]" + "[" + j + "]: ");
                mat2.setElement(i, j, sc.nextInt());
            }
        }
        System.out.println("The Matrix is " + ((isSymmetric(mat2) ? "Symmetric" : "Not Symm
    }
}
```

Output:

To Perform Matrix Multiplication **Code:**

```
import java.util.Scanner;
import matrix.Matrix;
class Multiplication
{
    static Matrix multiply(Matrix mat1, Matrix mat2)
    {
        if(mat1.getColumns() != mat2.getRows())
        {
            System.out.println("Cannot Multiply these matrices");
        }
        Matrix matMult = new Matrix(mat1.getRows(), mat2.getColumns());
        for(int i=0; i < mat1.getRows(); i++)
        {
            for(int j=0; j < mat1.getColumns(); j++)
            {
```

```
                for(int k = 0; k < mat1.getColumns(); k++)
                {
                    matMult.setElement(i, j, matMult.elementAt(i, j) + mat1.elementAt(i, k)
                }
            }
        }
        return matMult;
    }
    public static void main(String args[])
    {
        Scanner sc = new Scanner(System.in);
        System.out.print("Enter the no of rows of Matrix1: ");
        int rows = sc.nextInt();
        System.out.print("Enter the no of columns of Matrix1: ");
        int columns = sc.nextInt();
        Matrix mat1 = new Matrix(rows, columns);
        mat1.setMatrix();
        System.out.print("Enter the no of rows of Matrix2: ");
        rows = sc.nextInt();
        System.out.print("Enter the no of columns of Matrix2: ");
        columns = sc.nextInt();
        Matrix mat2 = new Matrix(rows, columns);
        mat2.setMatrix();
        System.out.print(multiply(mat1, mat2));
    }
}
```

Output:

Reverse the string and decide whether it is palindrome or not and Capitalize the String **Code:**

```
import java.util.Scanner;
class Pallindrome
{
    public static void main(String args[])
    {
        String in = "Hello";
        char str[] = in.toCharArray();
        char rev[] = new char[str.length];
        for(int i = 0; i < str.length; i++)
        {
            rev[i] = str[str.length - 1 - i];
        }
        System.out.println("The String is " + (in.equals(rev.toString()) ? "Pallindrome" :
        System.out.println("Capitalized String: " + in.toUpperCase());
    }
}
```

Output: