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

```java
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.compareTo("Hello")); // compareTo()
    System.out.println("str1 == \"Hello\": " + str1.equals("Hello"));        // equals()
    System.out.println( "str1 == \"Hello\" After trimming: " + str1.trim().equals("Hello")); // trim()
    System.out.println("str2.compareToIgnoreCase(\"world\"): " + str2.compareToIgnoreCase("world"));// compareToIgnoreCase()
    System.out.println("Compare str2.toLowerCase() and \"World\" ignoring the case: " + str2.toLowerCase().equalsIgnoreCase("World"));// toLowerCase() & equalsIgnoreCase()
    System.out.println("str3.toUpperCase(): " + str3.toUpperCase()); // toUpperCase()
    System.out.println("Replace First occurence of \"Java\" with \"Command Prompt\": "+ str5.replaceFirst("Java", "Command Prompt")); // replaceFirst()
   System.out.println("Replace All occurences of \"Java\" with \"Command Prompt\": " + str5.replaceAll("Java", "Command Prompt")); // replaceAll()
    System.out.println("Does str5.replaceAll(\"Java\", \"Command Prompt\") contains \"Java\": "+ str5.replaceAll("Java", "Command Prompt").contains("Java")); // contains()
    System.out.println("str5 ends with \"Java\": " + str5.endsWith("Java")); // endsWith()
    StringBuilder s = new StringBuilder(str3); // For using contentEquals which takes a CharSequence parameter
    System.out.println("Content of s equals content of str3: " + str3.contentEquals(s)); // contentEquals()
    System.out.println("\"\" is empty: " + "".isEmpty()); // isEmpty()
    System.out.println("Replace first occurence of I with O in str4: " + str4.replace('I', 'O')); // replace()
    System.out.println("Length of str5: " + str5.length()); // length()
    System.out.println("Character at index 3 in str5: " + str5.charAt(3)); // charAt()
    System.out.println("Substring of str5 from the index of where \"Java\" is found: " + 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.print("Replacing spaces with underscore in arr: ");
    System.out.print(arr);
  }
}
```

Output:

```
Returns 0 if str1 == "Hello": -40
str1 == "Hello": false
str1 == "Hello" After trimming: true
str2.compareToIgnoreCase("world"): 0
Compare str2.toLowerCase() and "World" ignoring the case: true
str3.toUpperCase: FROM JAVA
Replace First occurence of "Java" with "Command Prompt": From Command Prompt In Java
Replace All occurences of "Java" with "Command Prompt": From Command Prompt In Command Prompt
Does str5.replaceAll("Java", "Command Prompt") contains "Java": false
str5 ends with "Java": true
Content of s equals content of str3: true
"" is empty: true
Replace first occurence of I with O in str4: On Java
Length of str5: 17
Character at index 3 in str5: m
Substring of str5 from the index of where "Java" is found: Java In Java
Replacing spaces with underscore in arr: ____Hello
```

Matrix Class:

```java
// matrix/Matrix.java
package matrix;
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++)
    {
```

```java
      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:**

```java
// Symmetric.java
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(mat2);
    System.out.println("The Matrix is " + ((isSymmetric(mat2) ? "Symmetric" : "Not Symmetric")));
  }
}
```

Output:

```
Enter the order of the Matrix: 3
mat[0][0]: 1
mat[0][1]: 0
mat[0][2]: 0
mat[1][0]: 0
mat[1][1]: 1
mat[1][2]: 0
mat[2][0]: 0
mat[2][1]: 0
mat[2][2]: 1
1 0 0
0 1 0
0 0 1

The Matrix is Symmetric
```

To Perform Matrix Multiplication **Code:**

```java
// Multiplication.java
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 < mat2.getRows(); k++)
        {
          matMult.setElement(i, j, matMult.elementAt(i, j) + mat1.elementAt(i, k) * mat2.elementAt(k, j));
        }
      }
    }
    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.println("mat1 x mat2 = ");
    System.out.print(multiply(mat1, mat2));
  }
}
```

Output:

```
Enter the no of rows of Matrix1: 2
Enter the no of columns of Matrix1: 2
mat[0][0]: 1
mat[0][1]: 1
mat[1][0]: 1
mat[1][1]: 1
Enter the no of rows of Matrix2: 2
Enter the no of columns of Matrix2: 2
mat[0][0]: 1
```

```
mat[0][1]: 2
mat[1][0]: 1
mat[1][1]: 2
mat1 x mat2 =
2 4
2 4
```

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

```java
import java.util.Scanner;
class Pallindrome
{
  public static void main(String args[])
  {
    Scanner sc = new Scanner(System.in);
    System.out.print("Enter a String: ");
    String in = sc.next();
    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(new String(rev)) ? "Pallindrome" : "Not Pallindrome"));
    System.out.println("Capitalized String: " + in.toUpperCase());
  }
}
```

Output:

```
Enter a String: naman
The String is Pallindrome
Capitalized String: NAMAN
```