```java
// Define the Remote Interface (StringOperations.java):
import java.rmi.Remote;
import java.rmi.RemoteException;

public interface StringOperations extends Remote {
    boolean compareStrings(String str1, String str2) throws RemoteException;
    int countCharacters(String str) throws RemoteException;
}

// Implement the Remote Object (StringOperationsImpl.java):
import java.rmi.RemoteException;
import java.rmi.server.UnicastRemoteObject;

public class StringOperationsImpl extends UnicastRemoteObject implements StringOperations {
    public StringOperationsImpl() throws RemoteException {
        // Constructor
    }

    @Override
    public boolean compareStrings(String str1, String str2) throws RemoteException {
        return str1.equals(str2);
    }

    @Override
    public int countCharacters(String str) throws RemoteException {
        return str.length();
    }
}

// Create the RMI Server (StringServer.java):
import java.rmi.Naming;
import java.rmi.registry.LocateRegistry;

public class StringServer {
    public static void main(String[] args) {
        try {
            // Create the remote object
            StringOperations stringOperations = new StringOperationsImpl();

            // Create and start the RMI registry on port 1099
            LocateRegistry.createRegistry(1099);

            // Bind the remote object to a name in the RMI registry
            Naming.rebind("StringOperations", stringOperations);
```

```java
            System.out.println("Server is ready.");
        } catch (Exception e) {
            e.printStackTrace();
        }
    }
}
```

Create the RMI Client (StringClient.java):
```java
import java.rmi.Naming;
import java.util.Scanner;

public class StringClient {
    public static void main(String[] args) {
        try {
            // Look up the remote object by its name
            StringOperations stringOperations = (StringOperations)
Naming.lookup("rmi://localhost/StringOperations");

            // Get user input for strings
            Scanner scanner = new Scanner(System.in);
            System.out.print("Enter the first string: ");
            String str1 = scanner.nextLine();
            System.out.print("Enter the second string: ");
            String str2 = scanner.nextLine();

            // Compare strings
            boolean areEqual = stringOperations.compareStrings(str1, str2);
            System.out.println("The strings are equal: " + areEqual);

            // Count characters in a string
            int charCount = stringOperations.countCharacters(str1);
            System.out.println("Number of characters in the first string: " + charCount);

            // Close the scanner
            scanner.close();
        } catch (Exception e) {
            e.printStackTrace();
        }
    }
}
```

Compile and Run:
Compile all the Java source files:

javac StringOperations.java StringOperationsImpl.java StringServer.java StringClient.java

Start the RMI server in one terminal:
java StringServer
Run the RMI client in another terminal:
java StringClient