

## **Remote Interface:**

```
package RMI_5;

import java.rmi.Remote;
import java.rmi.RemoteException;

public interface PalindromeChecker extends Remote {
    boolean isPalindrome(String str) throws RemoteException;
}
```

## **Remote Interface Implementation:**

```
package RMI_5;

import java.rmi.RemoteException;
import java.rmi.server.UnicastRemoteObject;

public class PalindromeCheckerImpl extends UnicastRemoteObject implements
PalindromeChecker {
    protected PalindromeCheckerImpl() throws RemoteException {
        super();
    }

    @Override
    public boolean isPalindrome(String str) throws RemoteException {
        // Remove all non-alphanumeric characters and convert to lowercase
        String cleanStr = str.replaceAll("[^a-zA-Z0-9]", "").toLowerCase();

        // Check if the clean string is a palindrome
        int left = 0;
        int right = cleanStr.length() - 1;
        while (left < right) {
            if (cleanStr.charAt(left) != cleanStr.charAt(right)) {
                return false;
            }
            left++;
            right--;
        }
        return true;
    }
}
```

## **RMIServer:**

```
package RMI_5;

import java.rmi.RemoteException;
import java.rmi.registry.LocateRegistry;
import java.rmi.registry.Registry;
import java.rmi.server.UnicastRemoteObject;
```

```

public class RMIServer {
    public static void main(String[] args) {
        try {
            // Create an instance of the PalindromeCheckerImpl
            PalindromeCheckerImpl server = new PalindromeCheckerImpl();

            // Export the object to make it available as a remote object
            PalindromeChecker stub = (PalindromeChecker)
            UnicastRemoteObject.exportObject(server, 0);

            // Bind the remote object's stub in the registry
            Registry registry = LocateRegistry.createRegistry(1099);
            registry.bind("PalindromeService", stub);

            System.out.println("Server started...");
        } catch (Exception e) {
            System.err.println("Server exception: " + e.toString());
            e.printStackTrace();
        }
    }
}

```

### **RMIClient:**

```

package RMI_5;

import java.rmi.registry.LocateRegistry;
import java.rmi.registry.Registry;
import java.util.Scanner;

public class RMIClient {
    public static void main(String[] args) {
        try {
            // Get a reference to the RMI registry
            Registry registry = LocateRegistry.getRegistry("localhost", 1099);

            // Look up the remote object from the registry
            PalindromeChecker palindromeChecker = (PalindromeChecker)
            registry.lookup("PalindromeService");

            // Create a Scanner object for user input
            Scanner scanner = new Scanner(System.in);

            // Ask the user for input
            System.out.print("Enter a string or number to check for palindrome: ");
            String input = scanner.nextLine();

            // Call the remote method with the user's input
            boolean isPalindrome = palindromeChecker.isPalindrome(input);

            // Display the result
            System.out.println("'" + input + "' is a palindrome: " + isPalindrome);

            // Close the scanner

```

```
        scanner.close();
    } catch (Exception e) {
        System.err.println("Client exception: " + e.toString());
        e.printStackTrace();
    }
}
}
```