Experiment No:- 5

Program:

Palindrome Checker - :

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

/**
 * Remote interface for the Palindrome Checker service.
 * Defines methods that can be invoked remotely.
 */
public interface PalindromeChecker extends Remote {

    /**
     * Checks if a string is a palindrome.
     *
     * @param input The string to check
     * @return true if the string is a palindrome, false otherwise
     * @throws RemoteException if a remote communication error occurs
     */
    boolean isPalindrome(String input) throws RemoteException;

    /**
     * Checks if a number is a palindrome.
     *
     * @param number The number to check
     * @return true if the number is a palindrome, false otherwise
     * @throws RemoteException if a remote communication error occurs
     */
    boolean isPalindrome(int number) throws RemoteException;
}
```

Palindrome Server-

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

/**
 * Server implementation for the Palindrome Checker service.
 */
public class PalindromeServer implements PalindromeChecker {

    /**
     * Checks if a string is a palindrome.
     * A string is a palindrome if it reads the same backward as forward.
     */
    @Override
    public boolean isPalindrome(String input) throws RemoteException {
        if (input == null || input.isEmpty()) {
```

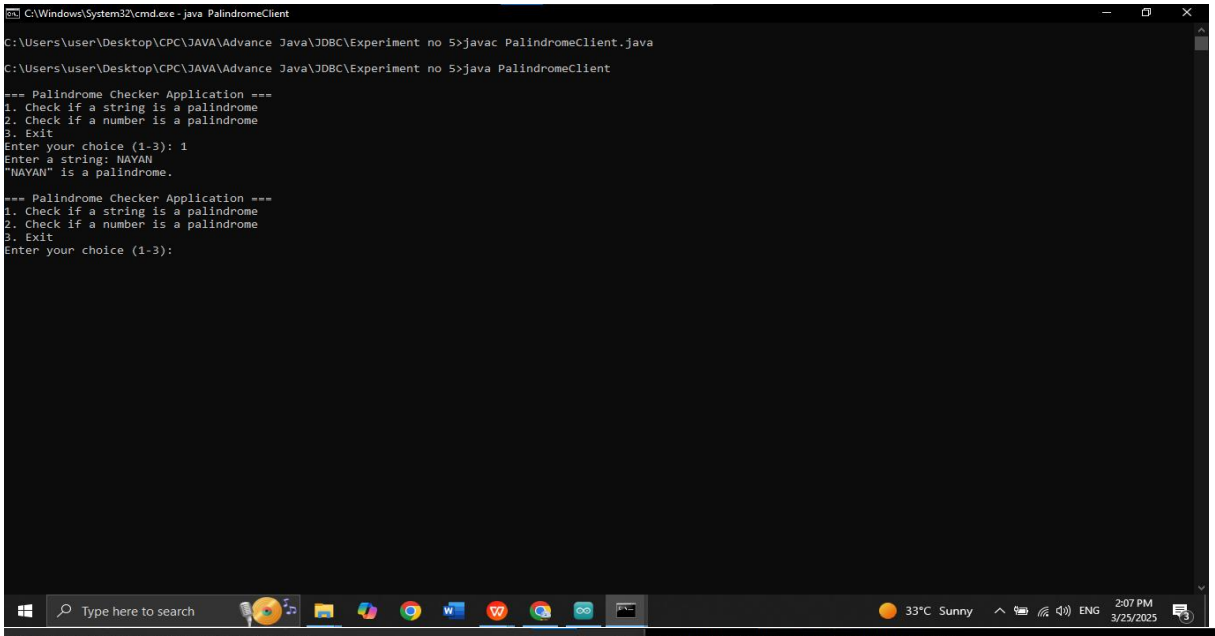
```
            return false;
        }

        // Remove spaces and convert to lowercase for more flexible
checking
        String cleanInput = input.replaceAll("\\s+", "").toLowerCase();
        int left = 0;
        int right = cleanInput.length() - 1;

        while (left < right) {
            if (cleanInput.charAt(left) != cleanInput.charAt(right)) {
                return false;
            }
            left++;
            right--;
        }

        return true;
    }

    /**
     * Checks if a number is a palindrome.
     * A number is a palindrome if it reads the same backward as forward.
     */
    @Override
    public boolean isPalindrome(int number) throws RemoteException {
        // Handle negative numbers
        if (number < 0) {
            return false; // Negative numbers can't be palindromes due to
the minus sign
        }

        // Convert to string and use the string palindrome checker
        return isPalindrome(Integer.toString(number));
    }

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

            // Export the remote object
            PalindromeChecker stub = (PalindromeChecker)
UnicastRemoteObject.exportObject(server, 0);

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

            // Bind the remote object's stub in the registry
            registry.rebind("PalindromeChecker", stub);

            System.out.println("Palindrome Server is running...");
            System.out.println("Press Ctrl+C to stop the server.");

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

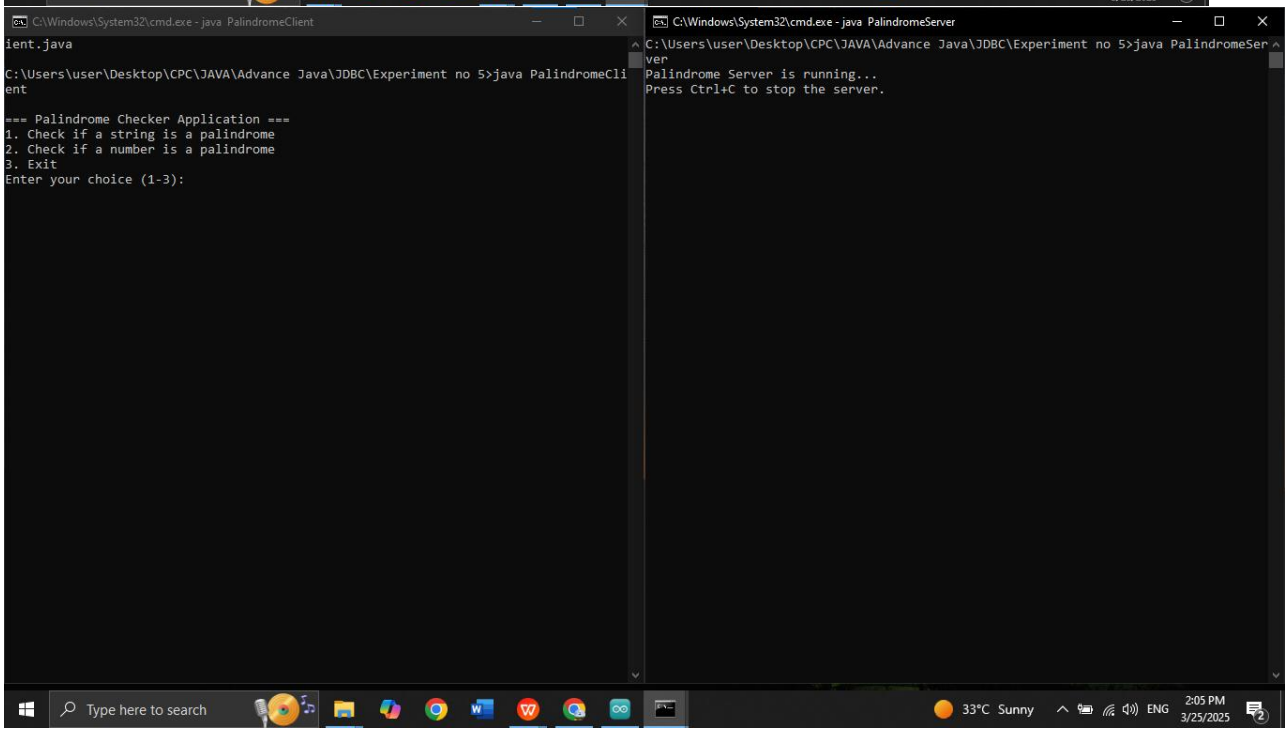
```
}
```

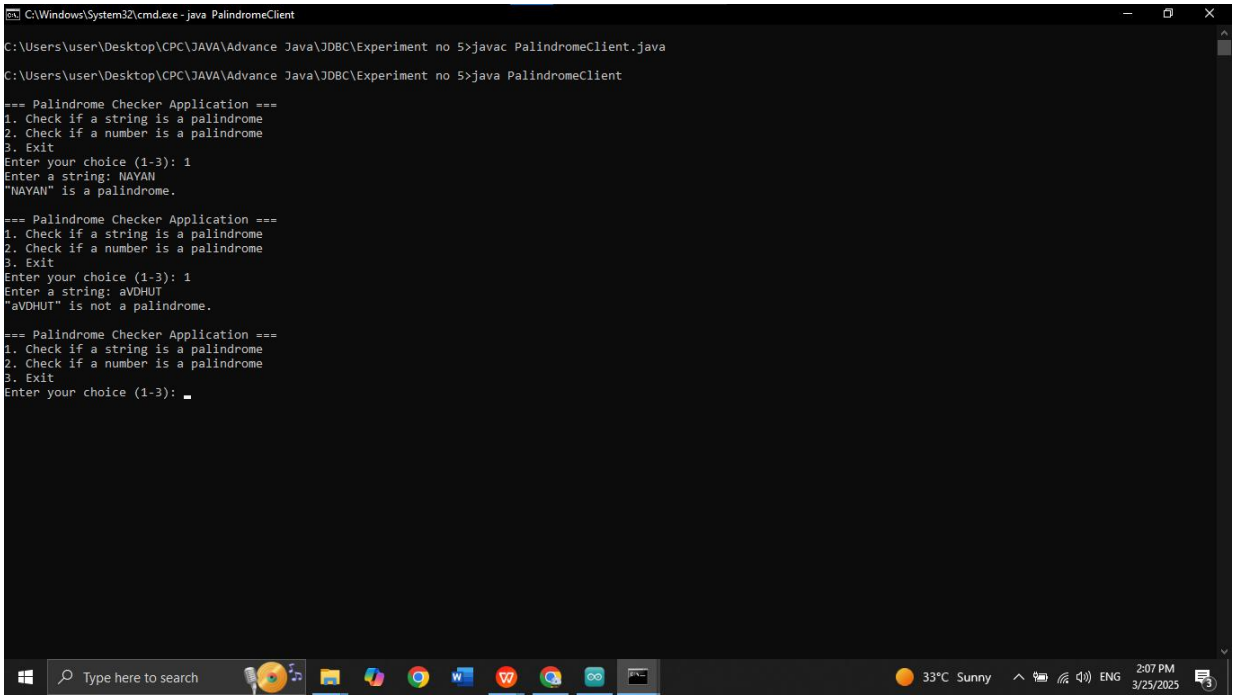
PalindromeClient-

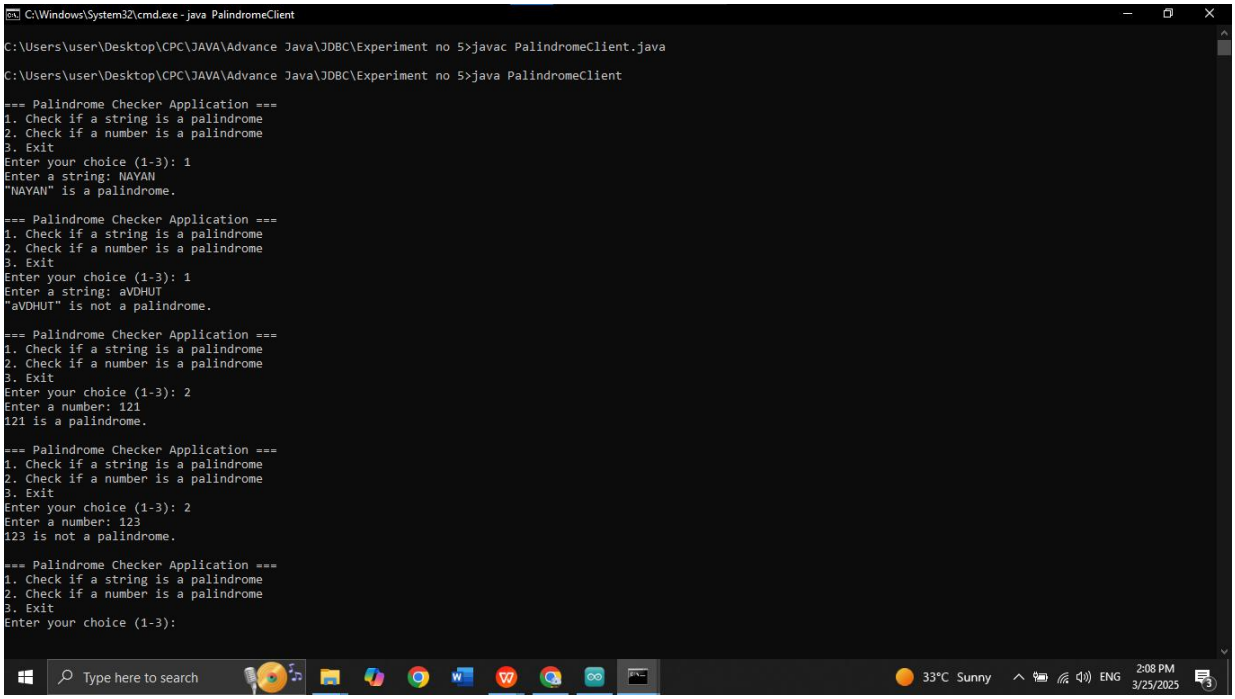
```
import java.rmi.registry.LocateRegistry;
import java.rmi.registry.Registry;
import java.util.Scanner;

/**
 * Client application for the Palindrome Checker service.
 */
public class PalindromeClient {

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

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

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

            while (true) {
                System.out.println("\n=== Palindrome Checker Application
===");
                System.out.println("1. Check if a string is a palindrome");
                System.out.println("2. Check if a number is a palindrome");
                System.out.println("3. Exit");
                System.out.print("Enter your choice (1-3): ");

                int choice = scanner.nextInt();
                scanner.nextLine(); // Consume newline

                switch (choice) {
                    case 1:
                        System.out.print("Enter a string: ");
                        String stringInput = scanner.nextLine();
                        boolean isStringPalindrome =
checker.isPalindrome(stringInput);

                        System.out.println("\"" + stringInput + "\" is " +
                                           (isStringPalindrome ? "a
palindrome." : "not a palindrome."));
                        break;

                    case 2:
                        System.out.print("Enter a number: ");
                        int numberInput = scanner.nextInt();
                        scanner.nextLine(); // Consume newline

                        boolean isNumberPalindrome =
checker.isPalindrome(numberInput);

                        System.out.println(numberInput + " is " +
```

```
                                            (isNumberPalindrome ? "a
palindrome." : "not a palindrome."));
                        break;

                    case 3:
                        System.out.println("Exiting the application...");
                        scanner.close();
                        return;

                    default:
                        System.out.println("Invalid choice. Please try
again.");
                }
            }

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


C:\Windows\System32\cmd.exe - java PalindromeClient
C:\Users\user\Desktop\CPC\JAVA\Advance Java\JDBC\Experiment no 5>javac PalindromeClient.java
C:\Users\user\Desktop\CPC\JAVA\Advance Java\JDBC\Experiment no 5>java PalindromeClient
=== Palindrome Checker Application ===
1. Check if a string is a palindrome
2. Check if a number is a palindrome
3. Exit
Enter your choice (1-3): 1
Enter a string: NAYAN
"NAYAN" is a palindrome.
=== Palindrome Checker Application ===
1. Check if a string is a palindrome
2. Check if a number is a palindrome
3. Exit
Enter your choice (1-3):
Type here to search
33°C Sunny
ENG
2:07 PM
3/25/2025


C:\Windows\System32\cmd.exe - java PalindromeClient
ient.java
C:\Users\user\Desktop\CPC\JAVA\Advance Java\JDBC\Experiment no 5>java PalindromeCli
ent
=== Palindrome Checker Application ===
1. Check if a string is a palindrome
2. Check if a number is a palindrome
3. Exit
Enter your choice (1-3):
C:\Windows\System32\cmd.exe - java PalindromeServer
C:\Users\user\Desktop\CPC\JAVA\Advance Java\JDBC\Experiment no 5>java PalindromeSer
ver
Palindrome Server is running...
Press Ctrl+C to stop the server.
Type here to search
33°C Sunny
ENG
2:05 PM
3/25/2025


C:\Windows\System32\cmd.exe - java PalindromeClient
C:\Users\user\Desktop\CPC\JAVA\Advance Java\JDBC\Experiment no 5>javac PalindromeClient.java
C:\Users\user\Desktop\CPC\JAVA\Advance Java\JDBC\Experiment no 5>java PalindromeClient
=== Palindrome Checker Application ===
1. Check if a string is a palindrome
2. Check if a number is a palindrome
3. Exit
Enter your choice (1-3): 1
Enter a string: NAYAN
"NAYAN" is a palindrome.
=== Palindrome Checker Application ===
1. Check if a string is a palindrome
2. Check if a number is a palindrome
3. Exit
Enter your choice (1-3): 1
Enter a string: aVDHUT
"aVDHUT" is not a palindrome.
=== Palindrome Checker Application ===
1. Check if a string is a palindrome
2. Check if a number is a palindrome
3. Exit
Enter your choice (1-3):
Type here to search
33°C Sunny
ENG
2:07 PM
3/25/2025


C:\Windows\System32\cmd.exe - java PalindromeClient
C:\Users\user\Desktop\CPC\JAVA\Advance Java\JDBC\Experiment no 5>javac PalindromeClient.java
C:\Users\user\Desktop\CPC\JAVA\Advance Java\JDBC\Experiment no 5>java PalindromeClient
=== Palindrome Checker Application ===
1. Check if a string is a palindrome
2. Check if a number is a palindrome
3. Exit
Enter your choice (1-3): 1
Enter a string: NAYAN
"NAYAN" is a palindrome.
=== Palindrome Checker Application ===
1. Check if a string is a palindrome
2. Check if a number is a palindrome
3. Exit
Enter your choice (1-3): 1
Enter a string: aVDHUT
"aVDHUT" is not a palindrome.
=== Palindrome Checker Application ===
1. Check if a string is a palindrome
2. Check if a number is a palindrome
3. Exit
Enter your choice (1-3): 2
Enter a number: 121
121 is a palindrome.
=== Palindrome Checker Application ===
1. Check if a string is a palindrome
2. Check if a number is a palindrome
3. Exit
Enter your choice (1-3): 2
Enter a number: 123
123 is not a palindrome.
=== Palindrome Checker Application ===
1. Check if a string is a palindrome
2. Check if a number is a palindrome
3. Exit
Enter your choice (1-3):
Type here to search
33°C Sunny
ENG
2:08 PM
3/25/2025