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

public interface BinarySearchInterface extends Remote {
   int binarySearch(int[] arr, int key) throws RemoteException;
}
```

2)
```
import java.rmi.RemoteException;
import java.rmi.server.UnicastRemoteObject;
import java.util.Arrays;

public class BinarySearchImpl extends UnicastRemoteObject implements BinarySearchInterface {

   protected BinarySearchImpl() throws RemoteException {
      super();
   }

   @Override
   public int binarySearch(int[] arr, int key) throws RemoteException {
      Arrays.sort(arr);  // Ensure the array is sorted before binary search
      int left = 0, right = arr.length - 1;

      while (left <= right) {
         int mid = left + (right - left) / 2;

         if (arr[mid] == key)
            return mid;  // Return the index if found

         if (arr[mid] < key)
            left = mid + 1;
         else
```

```
            right = mid - 1;
        }
        return -1; // Key not found
    }
}
```

3)
```
import java.rmi.registry.LocateRegistry;
import java.rmi.registry.Registry;

public class BinarySearchServer {
    public static void main(String[] args) {
        try {
            BinarySearchImpl searchObj = new BinarySearchImpl();
            Registry registry = LocateRegistry.createRegistry(1099);
            registry.rebind("BinarySearchService", searchObj);
            System.out.println("Binary Search RMI Server is running...");
        } catch (Exception e) {
            e.printStackTrace();
        }
    }
}
```

4)
```
import java.rmi.registry.LocateRegistry;
import java.rmi.registry.Registry;
import java.util.Scanner;

public class BinarySearchClient {
    public static void main(String[] args) {
        try {
            Registry registry = LocateRegistry.getRegistry("localhost", 1099);
```

```
        BinarySearchInterface searchObj = (BinarySearchInterface)
registry.lookup("BinarySearchService");

        Scanner scanner = new Scanner(System.in);
        System.out.print("Enter the number of elements: ");
        int n = scanner.nextInt();
        int[] arr = new int[n];

        System.out.println("Enter elements in sorted order:");
        for (int i = 0; i < n; i++) {
            arr[i] = scanner.nextInt();
        }

        System.out.print("Enter the key to search: ");
        int key = scanner.nextInt();
        scanner.close();

        int result = searchObj.binarySearch(arr, key);
        if (result != -1) {
            System.out.println("Element found at index: " + result);
        } else {
            System.out.println("Element not found.");
        }

      } catch (Exception e) {
        e.printStackTrace();
      }
    }
}
```