```java
package io.snyk.programming.tls;

import javax.net.ssl.SSLSocket;

import javax.net.ssl.SSLSocketFactory;

import java.io.*;


public class JavaTLSClientExample {


    private static final String[] protocols = new String[]{"TLSv1.3"};

    private static final String[] cipher_suites = new String[]{"TLS_AES_128_GCM_SHA256"};


    public static void main(String[] args) throws Exception {


        SSLSocket socket = null;

        PrintWriter out = null;

        BufferedReader in = null;

        try {

            // Step : 1

            SSLSocketFactory factory = (SSLSocketFactory) SSLSocketFactory.getDefault();


            // Step : 2

            socket = (SSLSocket) factory.createSocket("google.com", 443);


            // Step : 3

            socket.setEnabledProtocols(protocols);

            socket.setEnabledCipherSuites(cipher_suites);


            // Step : 4 {optional}

            socket.startHandshake();
```

```java
    // Step : 5
    out = new PrintWriter(
        new BufferedWriter(
            new OutputStreamWriter(
                socket.getOutputStream())));

    out.println("GET / HTTP/1.0");
    out.println();
    out.flush();

    if (out.checkError()) {
        System.out.println("SSLSocketClient:  java.io.PrintWriter error");
    }

    // Step : 6
    in = new BufferedReader(
        new InputStreamReader(
            socket.getInputStream()));

    String inputLine;
    while ((inputLine = in.readLine()) != null) {
        System.out.println(inputLine);
    }

} catch (Exception e) {
    e.printStackTrace();
} finally {
    if (socket != null) {
```

```java
            socket.close();

        }

        if (out != null) {

            out.close();

        }

        if (in != null) {

            in.close();

        }

      }

   }

}
```

_____

```java
import javax.net.ssl.SSLSocket;

import javax.net.ssl.SSLServerSocket;

import javax.net.ssl.SSLServerSocketFactory;

import java.io.*;


public class JavaTLSServerExample {


   private static final String[] protocols = new String[]{"TLSv1.3"};

   private static final String[] cipher_suites = new String[]{"TLS_AES_128_GCM_SHA256"};


   public static void main(String[] args) throws Exception {


      SSLServerSocket serverSocket = null;


      try {
```

```java
        // Step : 1
        SSLServerSocketFactory factory =
            (SSLServerSocketFactory) SSLServerSocketFactory.getDefault();


        // Step : 2
        serverSocket = (SSLServerSocket) factory.createServerSocket(8980);


        // Step : 3
        serverSocket.setEnabledProtocols(protocols);
        serverSocket.setEnabledCipherSuites(cipher_suites);


        // Step : 4
        SSLSocket sslSocket = (SSLSocket) serverSocket.accept();


        // Step : 5
        InputStream inputStream = sslSocket.getInputStream();
        InputStreamReader inputStreamReader = new
            InputStreamReader(inputStream);


        BufferedReader bufferedReader = new BufferedReader(inputStreamReader);
        String request = null;
        while((request = bufferedReader.readLine()) != null) {
            System.out.println(request);
            System.out.flush();
        }
    } catch (Exception e) {
        e.printStackTrace();
    } finally {
        if (serverSocket != null) {
```

```
            serverSocket.close();

        }

    }

  }

}
```