НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ УКРАЇНИ «КИЇВСЬКИЙ ПОЛІТЕХНІЧНИЙ ІНСТИТУТ ІМЕНІ ІГОРЯ СІКОРСЬКОГО»

ФАКУЛЬТЕТ ІНФОРМАЦІЙНО-ОБЧИСЛЮВАЛЬНИХ ТЕХНОЛОГІЙ

КАФЕДРА АВТОМАТИКИ ТА УПРАВЛІННЯ В ТЕХНІЧНИХ СИСТЕМАХ

**Лабораторна робота № 1**

**з предмету «Програмування комп’ютерних та віртуальних мереж» на тему «ЗНАЙОМСТВО З TCP SOCKETS»**

Виконала:

студентка групи ІТ-з03мп

Козак О. С.

Перевірив:

Галушко Д.О.

**Київ – 2020р.**

**Лабораторна робота №1**

**Тема: знайомство з TCP sockets**

**Мета: ознайомитися з принципами роботи TCP sockets.**

**Завдання:** Реалізувати клієнт-серверний застосунок:

**Хід роботи:**

**Лістинг коду:**

package CNVP;  
  
import java.io.\*;  
import java.net.InetAddress;  
import java.net.Socket;  
import java.util.Scanner;  
  
public class Main {  
 static int *port* = 5544;  
 static BufferedReader *reader* = null;  
 static BufferedWriter *writer* = null;  
  
 public static void sendReceive(Socket clientSocket, String input) {  
 try {  
 try {  
 *writer* = new BufferedWriter(new OutputStreamWriter(clientSocket.getOutputStream()));  
 *reader* = new BufferedReader(new InputStreamReader(clientSocket.getInputStream()));  
 *writer*.write(input+ "\n");  
 *writer*.flush();  
 String serverMassage = *reader*.readLine();  
 if (input.equals("get number of brigade")) {  
 System.*out*.println("number of my brigade: " + serverMassage);  
 } else if (input.equals("get surnames of brigade")) {  
 System.*out*.println("surnames of my brigade: " + serverMassage);  
 } else {  
 throw new IOException("Invalid operation");  
 }  
  
 } finally {  
 if (*reader* != null) *reader*.close();  
 if (*writer* != null) *writer*.close();  
 }  
  
 } catch (Exception e) {  
 e.printStackTrace();  
 }  
 }  
  
 public static void main(String[] args) {  
  
 String input = null;  
 Socket socketClient = null;  
 try {  
 try {  
 System.*out*.println("Input Ip-address");  
 Scanner in = new Scanner(System.*in*);  
 String ipAddress = in.nextLine();  
 InetAddress ipAddressOfServer = InetAddress.*getByName*(ipAddress);  
 while (true) {  
 input = in.nextLine();  
 if (input.equals("exit")) {  
 break;  
 }  
 socketClient = new Socket(ipAddressOfServer, *port*);  
 *sendReceive*(socketClient, input);  
 }  
 } finally {  
 if (socketClient != null) {  
 socketClient.close();  
 System.*out*.println("client is closed");  
 }  
 }  
 } catch (IOException ex) {  
 ex.printStackTrace();  
 }  
 }  
}