

## Client Server

### GIVE ME A QUESTION

Prosty program (w sumie dwa bo jeden kliencki a drugi serwer) pokazujący jak napisać aplikację komunikującą się z serwerem oraz serwer jak ma odpowiadać na informację od użytkownika.

Zacznijmy od servera:

Tworzymy klasę dla servera:

```
public class ServerConnection
```

[definiujemy zmienne globalne]

```
private ServerSocket serverSocket;  
private Socket clientSocket;  
private DataOutputStream out;  
private DataInputStream in;  
private int port;
```

Teraz tworzymy połączenie, a raczej serwer otwiera port i czeka na połączenie od klienta. Robimy to w dowolnej metodzie np.:

```
public void startConnection()
```

Otwieramy port (zazwyczaj wpisujemy liczbę większą niż 1024, poniżej porty są już zarezerwowane na inne usługi np.: 80 - www)

```
serverSocket = new ServerSocket(port);
```

Teraz pozwalamy na połączenie z zewnątrz (od klienta)

```
clientSocket = serverSocket.accept();
```

Przygotowujemy się na wysłanie danych do klienta oraz do odebranie danych od klienta

```
out = new DataOutputStream(clientSocket.getOutputStream());  
in = new DataInputStream(clientSocket.getInputStream());
```

Poniżej część kodu, który odpowiada za komunikację serwera z klientem

```

try {
    //Wysyłamy info do użytkownika który podłączy się do serwera
    out.writeUTF( str: "Jestem komputerem nowej generacji. Zadaaj pytanie, a na nie odpowiem!");
} catch (IOException e) {
    e.printStackTrace();
}

while(true){    //Inaczej nie będzie ciągłej komunikacji klient-server
    try {
        //czekamy na dane które prześle użytkownik
        input = in.readUTF();
        System.out.println("INPUT: "+input); //W konsoli na serwerze się wyświetli

        //Tutaj metoda która sprawdza jakie pytanie zadaliśmy i na tej podstawie generuje odpowiedź
        String answer = answerOnQuestion(input);

        /*
        Możesz też po prostu wpisać na stałe jakąś odpowiedź np:   String answer = "Tak to HAL!";
        Możesz też użyć Scanner aby wygenerować odpowiedź ręcznie po stronie serwera - wtedy powstanie chat
        String answer = sc.nextLine();
        */

        //Wysłanie do klienta danych
        out.writeUTF(answer);
        //Jeśli użytkownik wpisał quite - zakończenie pracy serwera
        //jeśli tego nie zrobimy serwer będzie dalej działał i czekał na następne połączenie
        if(input.equals("quite")) break;
    } catch (IOException e) {
        System.out.println("SOME PROBLEM "+e);
    }
}
}

```

Po stronie klienta sprawa ma się bardzo podobnie. Definiujemy zmienne globalnie.

```

private Socket socket;
private DataOutputStream out;
private DataInputStream in;
private String address;
private int port;

```

Przygotowanie do komunikacji z serwerem:

```

try{
    socket = new Socket(address, port);
    System.out.println("Connected");

    out = new DataOutputStream(socket.getOutputStream());
    in = new DataInputStream(socket.getInputStream());
}
catch(UnknownHostException u){
    System.out.println(u);
}
catch(IOException i){
    System.out.println(i);
}
}

```

### Komunikacja z serwerem:

```
String line = "", question="";
Scanner sc = new Scanner(System.in);
// keep reading until "Good bye" is receive from server
while (!line.equals("Good bye")){
    try{
        line = in.readUTF();
        System.out.println(line);
        if(!line.equals("Good bye")){
            question = sc.nextLine();
            out.writeUTF(question);
        }
    }
    catch(IOException i){
        System.out.println(i);
    }
}
```

Wyświetlenie danych od klienta

Dane z serwera różne od Good bye wtedy można wysłać dane do serwera

Wysłanie danych do serwera

### Oczywiście zamknięcie połączenia:

```
try{
    in.close();
    out.close();
    socket.close();
}
catch(IOException i){
    System.out.println(i);
}
```

## 3 zadania

1. Napisz aplikację klient / serwer, gdzie program klienta wysyła dwie liczby do serwera, który zwraca sumę, różnicę i iloczyn tych dwóch liczb.
2. Napisz aplikację klient/serwer, gdzie program klienta pobiera od użytkownika słowo. Słowo odczytaną z klawiatury wysyła do serwera. Server zapisuje słowa do Listy. Sprawdza czy podane słowo już zostało wpisane (wielkość liter nie ma znaczenia). Jeśli słowo już jest w liście wysyła komunikat – **Podane słowo już nam podaj inne**, w przeciwnym razie wysyła komunikat: **Dodano nowe słowo, jeśli chcesz wyświetlić listę słów wpisz: 1:wszystkie**.