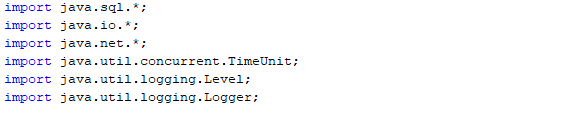
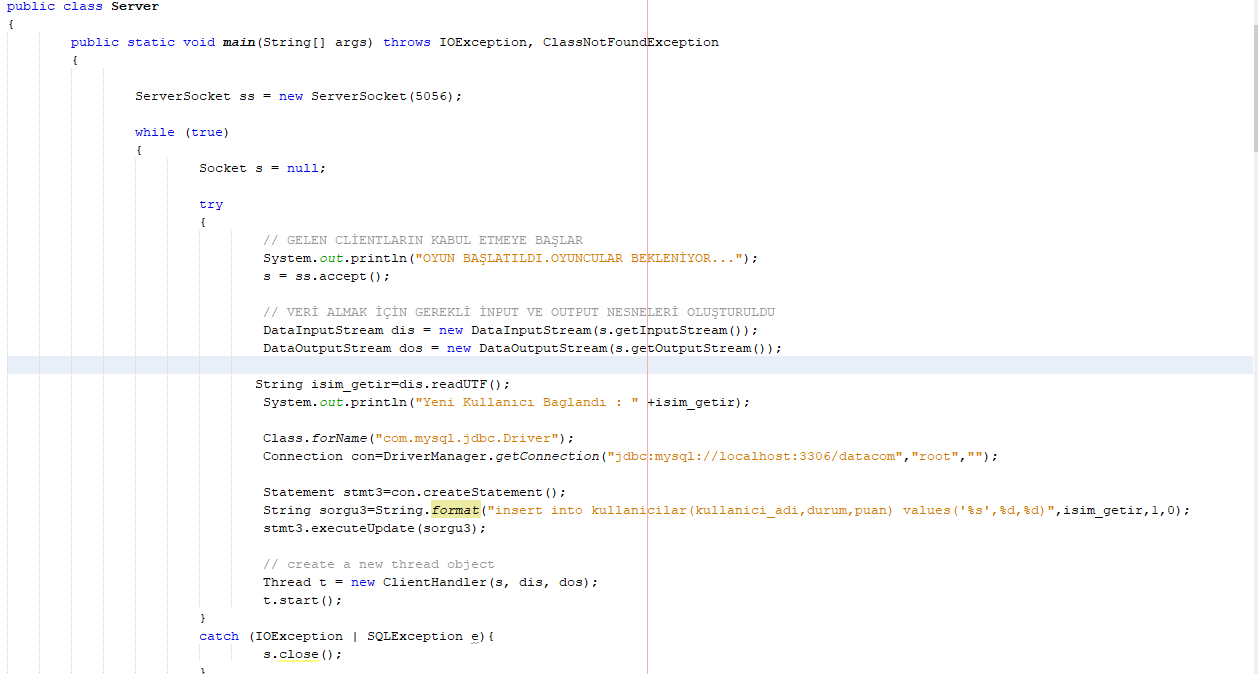
Projemde Java Programlama dilini ,Veri Tabanı Olarak Mysql’i kullandım.



Socket Programlamanın Java programlama dilinde duyduğu gerekli kütüphaneleri projeme ekledim.

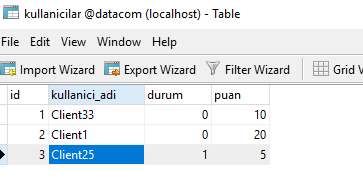
**SERVER CLASS**

Programımı ilk çalıştırdığında Main metodunun bulunduğu yerden başlayacağı için Javanın içinde ServerSocket sınıfından 5056 portu açarak Socket Programı başlattım.



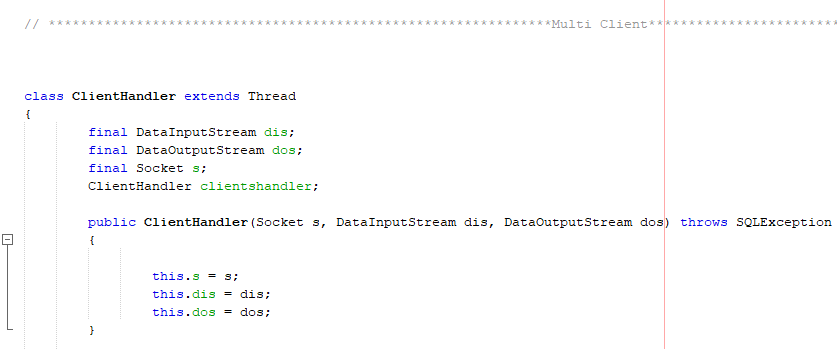
Accept Komutuyla Clientlarımı kabul ediyorum. Dis ve Dos Stream komutlarımızı oluşturduk. Dos Stream komutu Serverdan Clientlara Gönderilecek verileri için kullandığımız komut Dis komutu ise Clientlardan Gelen Cevapları aldığımız komuttur.

isim\_getir değişkeni Bağlanan Clientın adını alıyoruz. Class.forName kullanacağımız Sql belirtiyoruz.Connection komutuyla datacom adında veri tabanıma root kullanıcı adıyla bağlanıyorum.

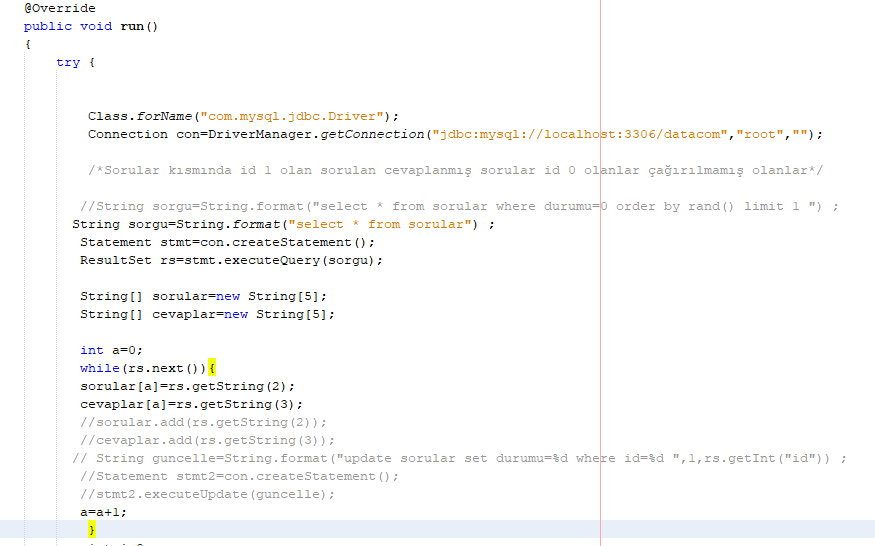


Clienttan aldığım ismi Veri Tabanına ekliyorum.Bu tabloda Clientın adını ,puanını ve durumunu kaydediyorum.

Therad() iş parçacığımı start ediyorum.

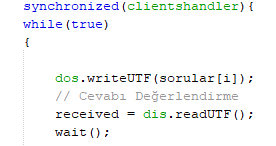


ClienHandler adında yeni bir class oluşturdum. Clienthandler classıma Javanın Thread objesini ekleyerek projeme MultiClient özelliği sağladım.

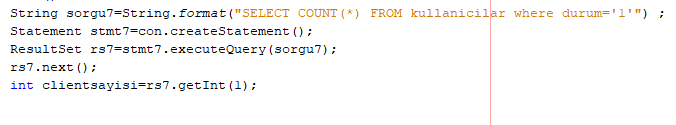


Run fonksiyonu Projeme Thread objesini ekleyince otomatik olarak geliyor.

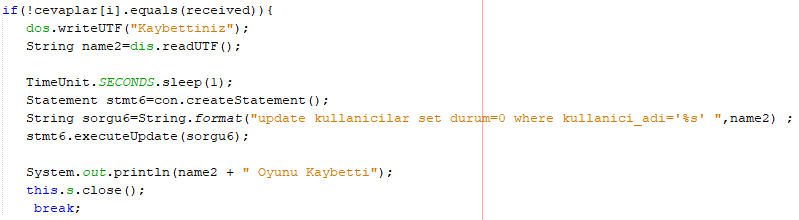
Run fonksiyonunda Kullanıclara Soru gönderme, Kullanıcıdan cevap alma , Soruları Değerlendirme, Kullanıcıların Aktiflik-Pasiflik durumu,Puan vb. durumları kayıt vb.işlemlerin yapıldığı fonksiyondur. Datacom adlı Veri Tabanıma bağlanarak Sorularımı ve Cevaplarımı oluşturmuş olduğum sorular ,cevaplar dizisine aktarıyorum.



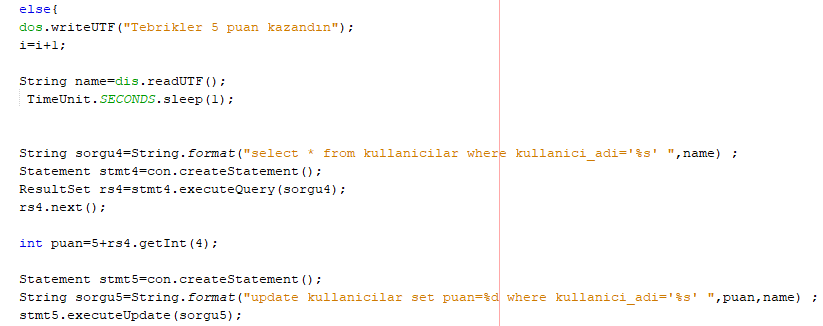
Sekronize-Wait komutuyla Sorularımın Kullanıcılara Tek tek gönderilmesini ve Cevapların bir kullanıcı cevap vermeden diğer kullanıcıya soruya gitmesini engellemektedir.**(Projemdeki Eksik olan kısmıda bu komutla Tamamladım Hocam.)**



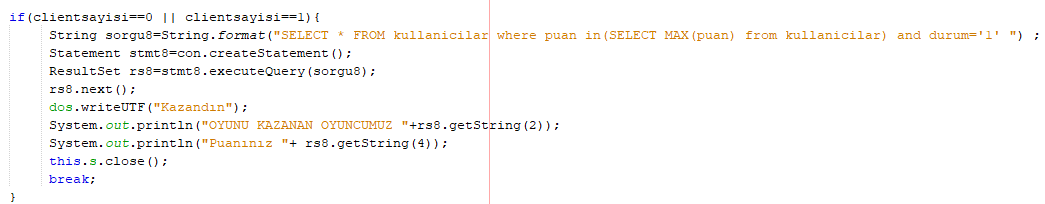
Veri tabanımdaki Aktif olan Kullanıcı Sayısı Clientsayısı Değişkenime Aktarıyorum.



Cevaplarıa kontrol Ettiriyorum. Cevap Yanlışsa Cliente “Kaybettiniz ” yazısı gönderiyorum. Cevap Yanlışsa Veri Tabanında Clientin durumunu “0” yapıyorum.Kullanıcı Socket programından çıkartıyorum.



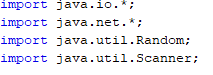
Client Doğru Cevap Verdiyse Clienta “Tebrikler 5 Puan Kazandın.”Mesajı yolluyorum.Veri tabanında kullanıcını puanına +5 paun ekliyorum.



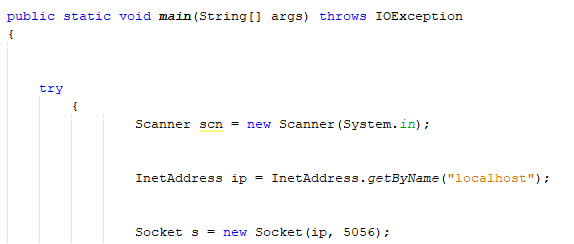
Client Sayısı 1 ve 0 kalınca Aktif kullanıcılardan puanı en yüksek olan kullanıcıyı Oyunu kazan kullanıcı olarak ilan ediyorum.Server Socketimi Sonlandırıyorum.

Oyun bu şekilde bitmiş oluyor.

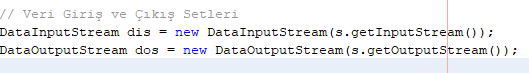
**CLİENT CLASS**



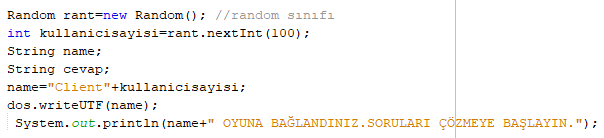
Client Classım için javadan gerekli kütüphalerimi projeme dahil ettim.



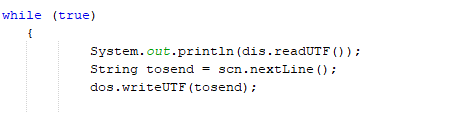
Kullanıcıdan Veriler almak scn adında bir değişken oluşturdum. Socket bağlantısı için İp adresi “127.0.0.1” olan portu 5056 açık olan Server Socket dosyasına s komutu ile bağlandım.



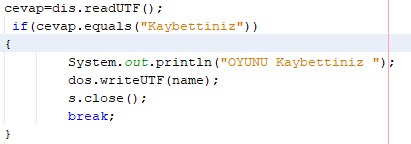
Veri Giriş ve Çıkışları için gerekli Stream dosylarımı projeme dahil ettim.



Random Sınıfı kullanarak Eşsiz bir ad elde etrim. Oluşturduğum Client adımı Server Soketime yolladım.



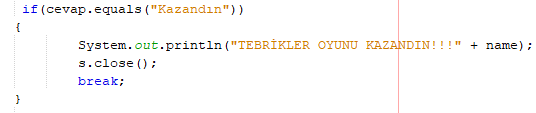
Server Socketimden gelen soruları ekrana yazdırıdım .Cevabımıda dos.writeUTF komutuyla değerlendirmesi içi Server Socketime gönderdim.



Cevabım Yanlışsa Server Socketten bana “Kaybettiniz Mesajı ” Geliyor. Ekrana “Oyunu Kaybettiniz” Yazdırıyorum. Clientimi Sonlandırıyorum.

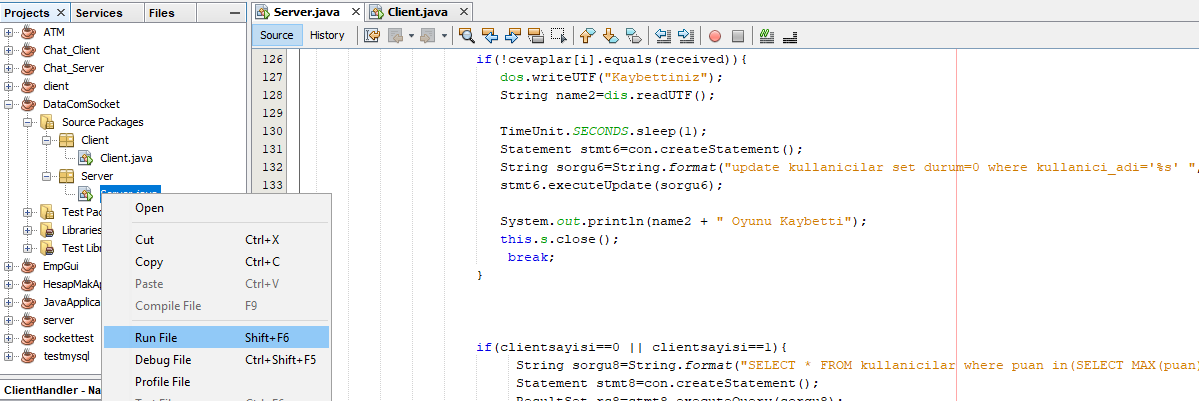


Değilse Diğer soruyu bekliyorum.

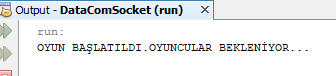


Serverdan “Kazandın” Mesajı alırsam.Oyunu Kazanan Client oluyor ve oturumunu kapatıyorum.

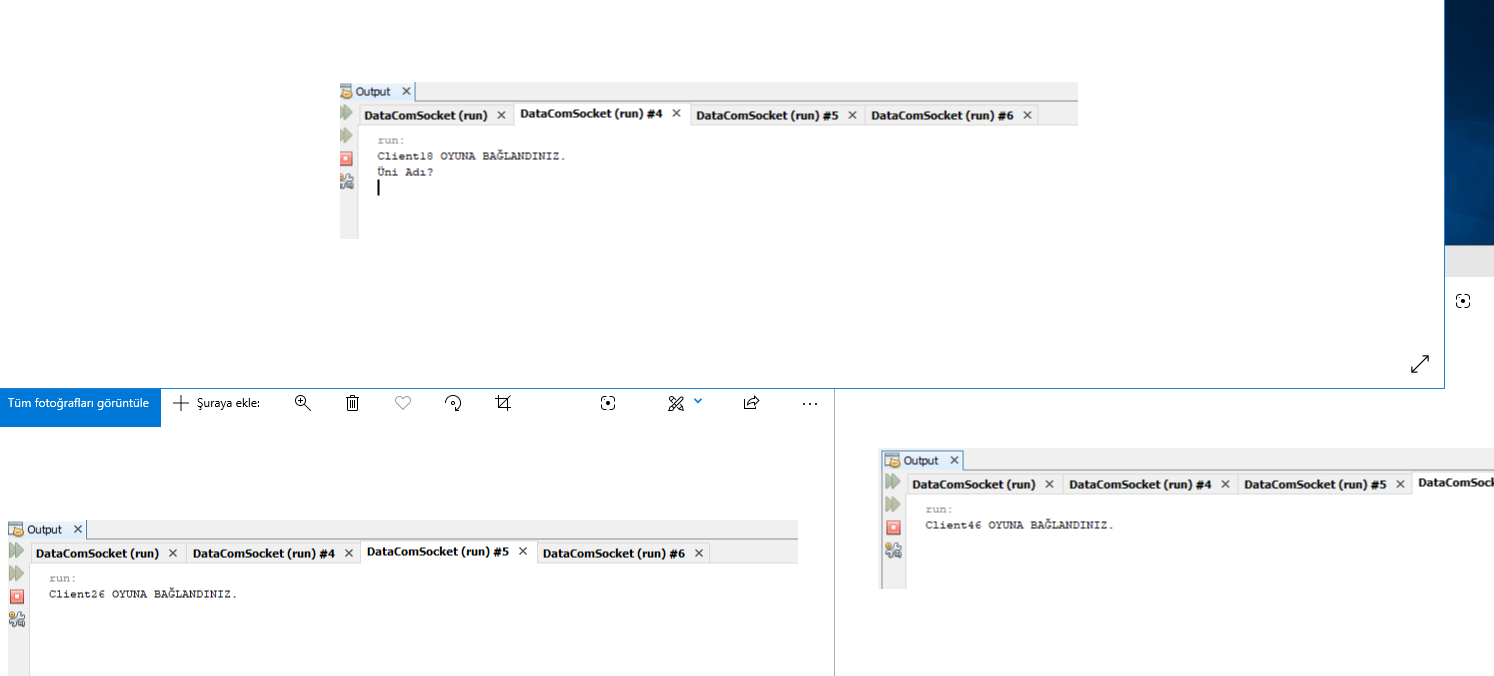
**PROJEMİN ÇALIŞTIRILMASI**



Projemi Netbeans’de Server.java dosyasına ve Client.java üzerine sağ tıklayarak “Run File” yapılması Gerekmektedir.

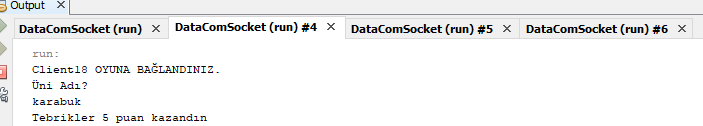


Server.java dosyamıza run yaptıktan sonra Client dosyamıza run yapmamız gerekmektedir.

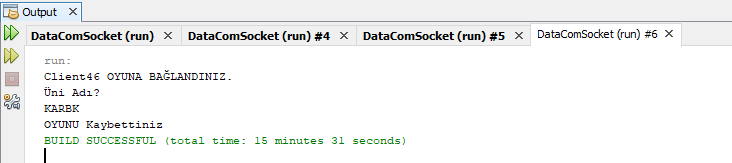


Clientleramıza Soru Gelme mantığı İlk Client Bağlandıktan sonra soru otomatik olarak 1.Sıradaki clientimize gelir.Diğer Clientler 1.Client Cevap verene Kadar diğer Clientlara Soru gitmez.İkinci ve Üçüncü Client içinde Böyledir

.Doğru Cevap Verdiğimizde 5 PUAN Kazanıp.Aşağıda görüldüğü gibi DataBase eklemektedir.

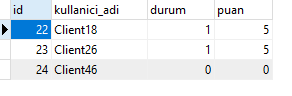


**KULLANICININ ELENMESİ DURUMU**



Client Yanlış Cevap Verdiği anda Oyunu Kaybeder ,Socketten Çıkarılıp.

.

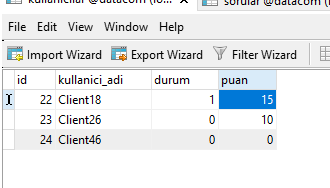
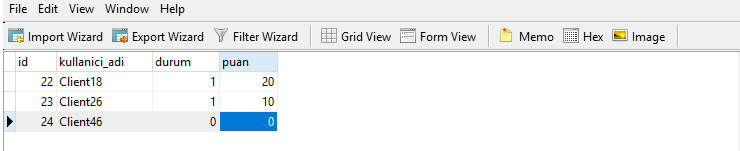


Databasede durumu 0(Pasif) olarak kaydedilir

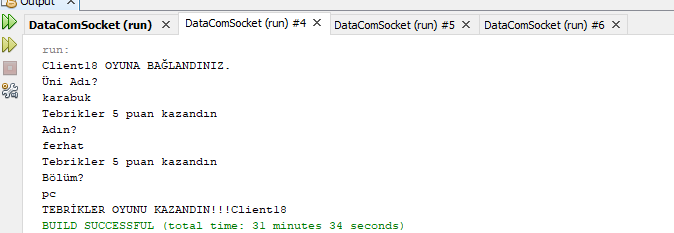
**SORULARIN BİTMESİNDE VEYA SON KULLANICI KALDIĞINDA KAZANAN**

Sorularımın Bitmesinde ise Kazanan Maksimum Puana Sahip Kullanıcıdır.Database Maksimum puana sahip Kullanıcı kimse o Kullanıcı kazanır.Eğer programda Son bir kullanıcı kaldıysa kazanan otomatikmen o kullanıcı olur

.(SORULARIN BİTMESİNDE KAZANAN CLİENT18 DATABASE DURMU)



(1 AKTİF KULLANICI KALDIĞINDA KAZANAN CLİENT18 DATABASE DURUMU



(KAZANAN KİŞİYE EKRANINA MESAJ YAZDIRILIR.)

**KAYNAK KOD**

**SERVER**

package Server;

import java.sql.\*;

import java.io.\*;

import java.net.\*;

import java.util.concurrent.TimeUnit;

import java.util.logging.Level;

import java.util.logging.Logger;

public class Server

{

public static void main(String[] args) throws IOException, ClassNotFoundException

{

ServerSocket ss = new ServerSocket(5056);

while (true)

{

Socket s = null;

try

{

// GELEN CLİENTLARIN KABUL ETMEYE BAŞLAR

System.out.println("OYUN BAŞLATILDI.OYUNCULAR BEKLENİYOR...");

s = ss.accept();

// VERİ ALMAK İÇİN GEREKLİ İNPUT VE OUTPUT NESNELERİ OLUŞTURULDU

DataInputStream dis = new DataInputStream(s.getInputStream());

DataOutputStream dos = new DataOutputStream(s.getOutputStream());

String isim\_getir=dis.readUTF();

System.out.println("Yeni Kullanıcı Baglandı : " +isim\_getir);

Class.forName("com.mysql.jdbc.Driver");

Connection con=DriverManager.getConnection("jdbc:mysql://localhost:3306/datacom","root","");

Statement stmt3=con.createStatement();

String sorgu3=String.format("insert into kullanicilar(kullanici\_adi,durum,puan) values('%s',%d,%d)",isim\_getir,1,0);

stmt3.executeUpdate(sorgu3);

// Theared Obl-jesi oluşturuyoruz

Thread t = new ClientHandler(s, dis, dos);

t.start();

}

catch (IOException | SQLException e){

s.close();

}

}

}

}

// \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*Multi Client\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

class ClientHandler extends Thread

{

final DataInputStream dis;

final DataOutputStream dos;

final Socket s;

ClientHandler clientshandler;

public ClientHandler(Socket s, DataInputStream dis, DataOutputStream dos) throws SQLException

{

this.s = s;

this.dis = dis;

this.dos = dos;

}

@Override

public void run()

{

try {

Class.forName("com.mysql.jdbc.Driver");

Connection con=DriverManager.getConnection("jdbc:mysql://localhost:3306/datacom","root","");

/\*Sorular kısmında id 1 olan sorulan cevaplanmış sorular id 0 olanlar çağırılmamış olanlar\*/

//String sorgu=String.format("select \* from sorular where durumu=0 order by rand() limit 1 ") ;

String sorgu=String.format("select \* from sorular") ;

Statement stmt=con.createStatement();

ResultSet rs=stmt.executeQuery(sorgu);

String[] sorular=new String[5];

String[] cevaplar=new String[5];

int a=0;

while(rs.next()){

sorular[a]=rs.getString(2);

cevaplar[a]=rs.getString(3);

//sorular.add(rs.getString(2));

//cevaplar.add(rs.getString(3));

// String guncelle=String.format("update sorular set durumu=%d where id=%d ",1,rs.getInt("id")) ;

//Statement stmt2=con.createStatement();

//stmt2.executeUpdate(guncelle);

a=a+1;

}

int i=0;

String received = null;

synchronized(clientshandler){

while(true)

{

dos.writeUTF(sorular[i]);

// Cevabı Değerlendirme

received = dis.readUTF();

wait();

String sorgu7=String.format("SELECT COUNT(\*) FROM kullanicilar where durum='1'") ;

Statement stmt7=con.createStatement();

ResultSet rs7=stmt7.executeQuery(sorgu7);

rs7.next();

int clientsayisi=rs7.getInt(1);

if(!cevaplar[i].equals(received)){

dos.writeUTF("Kaybettiniz");

String name2=dis.readUTF();

TimeUnit.SECONDS.sleep(1);

Statement stmt6=con.createStatement();

String sorgu6=String.format("update kullanicilar set durum=0 where kullanici\_adi='%s' ",name2) ;

stmt6.executeUpdate(sorgu6);

System.out.println(name2 + " Oyunu Kaybetti");

this.s.close();

break;

}

if(clientsayisi==0 || clientsayisi==1){

String sorgu8=String.format("SELECT \* FROM kullanicilar where puan in(SELECT MAX(puan) from kullanicilar) and durum='1' ") ;

Statement stmt8=con.createStatement();

ResultSet rs8=stmt8.executeQuery(sorgu8);

rs8.next();

dos.writeUTF("Kazandın");

System.out.println("OYUNU KAZANAN OYUNCUMUZ "+rs8.getString(2));

System.out.println("Puanınız "+ rs8.getString(4));

this.s.close();

break;

}

else{

dos.writeUTF("Tebrikler 5 puan kazandın");

i=i+1;

String name=dis.readUTF();

TimeUnit.SECONDS.sleep(1);

String sorgu4=String.format("select \* from kullanicilar where kullanici\_adi='%s' ",name) ;

Statement stmt4=con.createStatement();

ResultSet rs4=stmt4.executeQuery(sorgu4);

rs4.next();

int puan=5+rs4.getInt(4);

Statement stmt5=con.createStatement();

String sorgu5=String.format("update kullanicilar set puan=%d where kullanici\_adi='%s' ",puan,name) ;

stmt5.executeUpdate(sorgu5);

}

}

// }

} catch (SQLException | ClassNotFoundException | IOException | InterruptedException ex) {

Logger.getLogger(ClientHandler.class.getName()).log(Level.SEVERE, null, ex);

}

}

**CLİENT**

package Client;

import java.io.\*;

import java.net.\*;

import java.util.Random;

import java.util.Scanner;

// Client class

public class Client

{

public static void main(String[] args) throws IOException

{

try

{

Scanner scn = new Scanner(System.in);

InetAddress ip = InetAddress.getByName("localhost");

Socket s = new Socket(ip, 5056);

// Veri Giriş ve Çıkış Setleri

DataInputStream dis = new DataInputStream(s.getInputStream());

DataOutputStream dos = new DataOutputStream(s.getOutputStream());

Random rant=new Random(); //random sınıfı

int kullanicisayisi=rant.nextInt(100);

String name;

String cevap;

name="Client"+kullanicisayisi;

dos.writeUTF(name);

System.out.println(name+" OYUNA BAĞLANDINIZ.");

while (true)

{

System.out.println(dis.readUTF());

String tosend = scn.nextLine();

dos.writeUTF(tosend);

cevap=dis.readUTF();

if(cevap.equals("Kaybettiniz"))

{

System.out.println("OYUNU Kaybettiniz ");

dos.writeUTF(name);

s.close();

break;

}

if(cevap.equals("Kazandın"))

{

System.out.println("TEBRİKLER OYUNU KAZANDIN!!!" + name);

s.close();

break;

}

else

dos.writeUTF(name);

System.out.println(cevap);

}

// Bağlantıları kapatma

scn.close();

dis.close();

dos.close();

}catch(IOException e){

}

}

**KAYNAKÇA**

* <https://docs.oracle.com/javase/tutorial/networking/sockets/index.html>
* <https://www.geeksforgeeks.org/socket-programming-in-java/>
* <https://www.javatpoint.com/example-to-connect-to-the-mysql-database>
* <http://bilgisayarkavramlari.sadievrenseker.com/2009/11/20/java-ile-sohbet-istemcisunucusu/>
* <https://www.it-swarm.net/tr/java/javada-nasil-gecikme-yapabilirim/1047231861/>
* <https://www.yazilimkodlama.com/java/java-mysql-veritabani-baglantisi-select-insert-update-delete/>