```
/*
CS2524: DISTRIBUTED SYSTEMS AND SECURITY
ASSESSMENT MUD GAME
WRITTEN BY BRADLEY SCOTT
B.SCOTT.16@ABERDEEN.AC.UK
STUDENT ID: 51661169



*/




package cs3524.solutions.mud;


import java.rmi.Naming;
import java.rmi.RMISecurityManager;
import java.net.InetAddress;
import java.util.Iterator;
import java.rmi.server.UnicastRemoteObject;
import java.io.InputStreamReader;
import java.util.List;
import java.io.BufferedReader;

import java.util.Scanner;
/*
command to run me:

java cs3524.solutions.mud.MUDclient <hostname> <registry port>
java cs3524.solutions.mud.MUDclient pug-MacBookPro 50010



*/

public class MUDclient {
        static MUDinterface service;

        static BufferedReader in = new BufferedReader( new InputStreamReader( System.in )
);
    private static String username;
    private static String location;
    private static String MUDname;
    private static String newlocation;


        public static void main(String args[]) throws Exception {
                // Check for missing arguments
                if(args.length < 2) {
                        System.err.println("Missing arguments. Please specify both <host>
 <port>");
                        return;
                }

                String hostName = args[0];
                int port = Integer.parseInt(args[1]);

                try{
                        //Create registration URL from hostname, port
                        String regUrl = "rmi://" + hostName + ":" + port + "/Mudservice";
                        service = (MUDinterface)Naming.lookup(regUrl);

            System.out.println("client has connected to server officially");
                        start();
                }
                catch (java.io.IOException e) {
                        System.err.println("There has been an input error!");
                        System.err.println(e.getMessage());
                }
    }


    static void start() throws Exception{
        System.out.print(service.welcome());

        MUDname = in.readLine();
                try
                {
            if (service.pickMUD(MUDname).equals("False")){
                System.out.print("please enter an existing MUD name ");
                start();
            }

                        System.out.print(service.pickMUD(MUDname));


                        //ask user for username
            username = in.readLine();
            //add user to user list
                        service.addUser(username);



            // call method that begins gameplay
                        play();
                }
                catch(Exception e)
                {
                        System.out.println("server is down ");
        }
    }





    static void play() throws Exception {
    //gameplay variables
    Scanner reader = new Scanner(System.in);  // Reading from System.in
    String move = "";
    boolean play = true;







    location = service.myStartLocation();

    service.addThing(location, username);
    System.out.println("//////////MUD GAME IS BELOW /////////");
    System.out.println("start location is: " + location);

    //game is here

    try {
        while(play){

            System.out.print("\ntype help for a list of available commands \nplease enter
 a command >  ");
            move = reader.nextLine(); // Scans the next token of the input as an string.
            if (move.equalsIgnoreCase("north")||move.equalsIgnoreCase("east")||move.equal
sIgnoreCase("south")||move.equalsIgnoreCase("west")){
```

```
                location = service.moveThing(location, move.toLowerCase(), username);

                System.out.println("after moving " + move + " new location is " + locatio
n);
            }
            if (move.equalsIgnoreCase("look")) {
                System.out.println(service.locationInfo(location));

            }
            if(move.equalsIgnoreCase("online")){
                System.out.println(service.whoIsonline());
            }
            if (move.equalsIgnoreCase("take")) {

                System.out.println(service.ItemsAtLocation(location));
                System.out.print("what item would you like to take? >");
                String item = reader.nextLine();
                Boolean pickup = service.take(item, location);
                if (pickup){
                    System.out.println("item taken successfully and added to inventory");

                }
                else {
                    System.out.println("item could not be tatken make sure spelling is co
rrect");
                }

            }
            if(move.equalsIgnoreCase("make mud")){
                System.out.print("please enter a name for your new mud > ");
                String nameofMUD = reader.nextLine();
                service.makeMUD(nameofMUD);
            }

            if(move.equalsIgnoreCase("change mud")){
                service.delThing(location, username);
                service.removeUser(username);
                start();
            }

            if(move.equalsIgnoreCase("help")){
                System.out.println("\n list of available commands are as follows: \n");
                System.out.println(" \n //////// movement /////// \n");
                System.out.println(" typing \"north\" will move your character north if p
ossible \n");
                System.out.println(" typing \"east\" will move your character east if pos
sible \n");
                System.out.println(" typing \"south\" will move your character south if p
ossible \n");
                System.out.println(" typing \"west\" will move your character west if pos
sible \n");
                System.out.println(" \n //////// other gameplay commands /////// \n");
                System.out.println("typing \"look\"will display all users and items in yo
ur current location \n");
                System.out.println("typing \"take\" will present you with items you can a
dd to your inventory \n");
                System.out.println("typing \"online\"will display all users online the cu
rrent mud \n");
                System.out.println(" \n //////// MUD world commands /////// \n");
                System.out.println(" typing \"make mud\" will enable you to create a new
MUD world");
                System.out.println(" typing \"change mud\" will exit your current mud and
 allow you to enter another mud world");
                System.out.println(" typing \"quit\" will quit the game");
            }

            if(move.equalsIgnoreCase("quit")){
                //remove user from list
                service.delThing(location, username);
                service.removeUser(username);
                System.out.println("quitting game");
                System.exit(0);
            }




            //once finished


        }
    }
    catch(Exception e) {
        return;
    }







     }


}
```