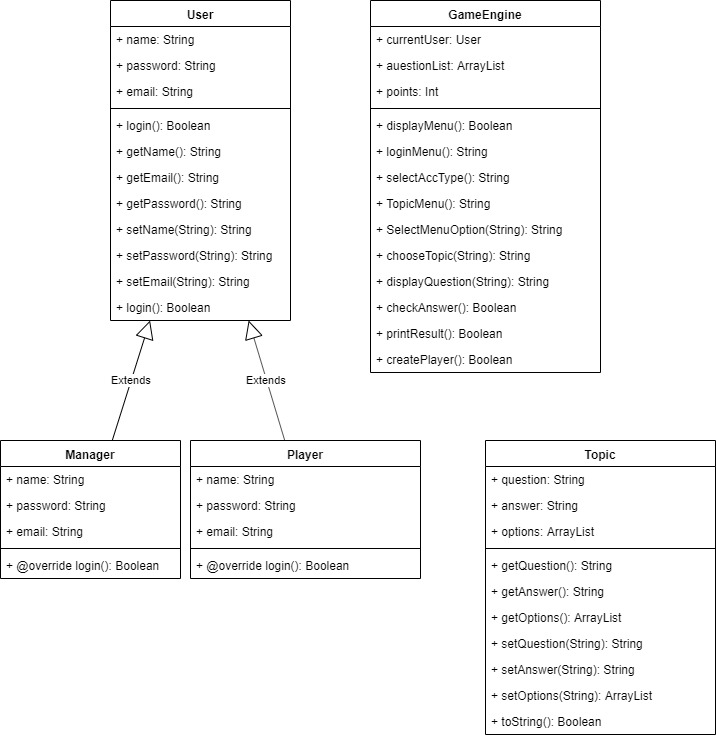
Java Project Trivia Game

This is OOP on a Trivia Game using Java

I have created a class diagram for my program which changed alot throughout but I came with this final take on my class Diagram.

# Class Diagram



#### Here I created a class Diagram in which I have 6 class

* 1st class Game Engine{} class which is core of Game Every thing Happens in this class.
* 2nd class User{} Is where I am passing and creating Fields and attributes

And class player{} extends from class User.so both can use same method using inheritance

In both class there is UserId,Password.

* 3rd class Main{} In this there is only main menu in that user will choose Option.
* 4th class Topic{] in there is 3 topic I added for game in that I Used files and I am reading files with array index and accessing Files.
* 5th class is manager{} extends from class user in this manager can log in and play game I was not Able to complete editing part but manage can play and see files.

## How I worked on this.

At first I did this manually we can say hardcoding and later I did it with the Array during this period I had a special class for all 4 question type beacuase I was not able to bind as one.Later when I started working as a File System I saw I can Bind it all as one class and function and later call the function to the main class a paramater for the file as a string.

Hard Part

The hard part of working on the projects comes when I started woking on the edit part for admin here I made a function to change all kind of lines as I made a fucntion which changes the first line where I am writing the total number of question then I ask the admin to enter 6 and later 6 string which should make them write 1.question 4.options 1.answer this make up the 6 string.later I change the question number which I asked before to the admin so bsically it changes the first line which says how many questions are there which is very important for the file sysmtem for this program.

# Algorithm

1. Start
2. Choose Option-Log in Or Register
3. If log in then Choose As Admin or Player

Else register new player

1. If Correct User ID and Password then Show Menu
2. Choose Type of question you want to play
3. If sport chosen then show Sports question

Else if show movie question

Else General knowledge

1. User can choose answer from Options
2. If Answer is right then it will increase 1 point

And add in total points

1. It will show Percentage of your score
2. Exit

# Classes

Main Class

**public** **class** Main {

**public** **static** **void** main(String[] args) {

// Display menu

Game\_Engine.*DisplayMenu*();

//Player p=new Player();

**int** choice = Game\_Engine.*SelectMenuOptions*();

**switch**(choice) {

**case** 1:

Game\_Engine.*logInMenu*();

**break**;

**case** 2:

Game\_Engine.*createPlayer*();

**break**;

**case** 3:

System.***out***.println("Thank you for playing Trivia!");

**break**;

**default**:

**break**;

}

}

}

**Game Engine**

import java.io.IOException;

import java.util.ArrayList;

import java.util.Scanner;

public class Game\_Engine {

private static Scanner input;

private static User currentUser;

private static ArrayList<Topic> questionsList = new ArrayList<>();

private static int points = 0;

//Menu Of Main Page

public static void DisplayMenu() {

System.out.println("Welcome to Trivia World");

System.out.println("1 : Log In");

System.out.println("2 : Register");

System.out.println("3 : Exit");

System.out.println("Select:");

}

//Select Login User

public static void logInMenu() {

System.out.println("1 : Log-in as Manager");

System.out.println("2 : Log-in as Player");

System.out.println("Select: ");

selectAccType(SelectMenuOptions());

}

//Select the User is Manager or Player

public static void selectAccType(int choice) {

switch(choice) {

case 1:

asManager();

break;

case 2:

asPlayer();

break;

default:

System.out.println("Select from above menu");

logInMenu();

break;

}

}

//Display Player Menu

private static void asPlayer() {

System.out.println("\*Loged-in as Player\*");

boolean created = false;

currentUser = new Player();

try {

created = currentUser.login();

} catch (IOException e) {

// TODO Auto-generated catch block

e.printStackTrace();

}

if(created) {

TopicMenu();

}

}

//Display Manager Menu

private static void asManager() {

System.out.println("\*Loged-in as Manager\*");

boolean created = false;

currentUser = new Manager();

try {

created = currentUser.login();

} catch (IOException e) {

// TODO Auto-generated catch block

e.printStackTrace();

}

if(created) {

//TopicMenu();

}

}

//Topic Menu

private static void TopicMenu() {

System.out.println("Select Genre you would like to play");

System.out.println("1 : General Knowledge");

System.out.println("2 : Sports");

System.out.println("3 : Movies");

System.out.println("Select :");

chooseTopic(SelectMenuOptions());

}

//Select User

public static int SelectMenuOptions() {

input = new Scanner(System.in);

int choice = input.nextInt();

return choice;

}

//Choosing Topic using Switch

private static void chooseTopic(int choice) {

Topic tempTopic = new Topic();

switch(choice) {

case 1:

try {

questionsList = tempTopic.getQuestions("gk");

} catch (IOException e) {

// TODO Auto-generated catch block

e.printStackTrace();

}

break;

case 2:

try {

questionsList = tempTopic.getQuestions("sp");

} catch (IOException e) {

// TODO Auto-generated catch block

e.printStackTrace();

}

break;

case 3:

try {

questionsList = tempTopic.getQuestions("mov");

} catch (IOException e) {

// TODO Auto-generated catch block

e.printStackTrace();

}

break;

default:

TopicMenu();

break;

}

if(!questionsList.isEmpty()) {

displayQuestions(0);

}

}

private static void displayQuestions(int index){

System.out.println(questionsList.get(index));

System.out.println("Select :");

checkAnswer(questionsList.get(index).answer,SelectMenuOptions(),index);

}

private static void checkAnswer(String option ,int ans, int index) {

if(Integer.toString(ans).equals(option)) {

System.out.println("Correct ans ");

System.out.println("Current Score : " + ++points);

if(questionsList.size() != index+1) {

displayQuestions(++index);

}else {

printResult(points);

}

}else {

System.out.println("Wrong ans ");

System.out.println("Current Score : " + points);

if(questionsList.size() != index+1) {

displayQuestions(++index);

}else {

printResult(points);

}

}

}

private static void printResult(int points) {

System.out.println("Your Total Score is : " + points);

}

//Creating Player When you select register

public static void createPlayer() {

Player tempPlayer = new Player();

boolean created = false;

try {

created = tempPlayer.createPlayer();

} catch (IOException e) {

e.printStackTrace();

}

if(created){

currentUser = tempPlayer;

TopicMenu();

}

}

}

### Class Manager

package p1;

import java.io.IOException;

import java.util.Scanner;

public class Manager extends User {

public Manager() {

super("admin", "admin", "admin");

}

//Using override Method to log in manager And extending user class

@Override

public boolean login() throws IOException {

System.out.println(this.Email);

Scanner input = new Scanner(System.in);

System.out.println("Enter email : ");

String email = input.next();

if(this.Email.equals(email)) {

System.out.println("Enter pass : ");

String password = input.next();

if(this.Password.equals(password)) {

System.out.println("login");

return true;

}else {

System.out.println("wrong password ");

try {

this.login();

} catch (Exception e) {

// TODO: handle exception

}

}

}else {

System.out.println("wrong user ");

try {

this.login();

} catch (Exception e) {

// TODO: handle exception

}

}

return false;

}

}

### Class Player

package p1;

import java.util.Scanner;

import java.util.regex.Pattern;

import java.io.BufferedReader;

import java.io.BufferedWriter;

import java.io.FileReader;

import java.io.FileWriter;

import java.io.IOException;

//Using User class as parent

public class Player extends User {

public Player(String Name, String password, String email) {

super(Name, password, email);

}

public Player() {

}

//Create Player

public boolean createPlayer() throws IOException {

Scanner input = new Scanner(System.in);

String line;

BufferedReader fileReader = new BufferedReader(new FileReader("c:/temp/players.txt"));

System.out.println("Enter Name : ");

this.Name = input.next();

System.out.println("Enter email : ");

this.Email = input.next();

System.out.println("Enter pass : ");

this.Password = input.next();

//Split Player Information From file

while ((line = fileReader.readLine()) != null) {

String[] player = line.split(Pattern.quote(","));

this.allPlayers.add(new Player(player[0],player[2],player[1]));

}

//Check if player is already there or not

boolean found = false;

for (Player player : this.allPlayers)

{

if(this.Email.equals(player.Email)) {

found = true;

break;

}

}

//Creating Player and save in File

if(!found) {

BufferedWriter fileWriter = new BufferedWriter(new FileWriter("c:/temp/players.txt",true));

String playerInfo = this.Name + "," + this.Email + "," + this.Password;

fileWriter.write(playerInfo+"\n");

fileWriter.close();

return true;

}else {

System.out.println("Player Already Exists.Try again With Uniqe EmailID");

this.createPlayer();

}

return false;

}

@Override

public boolean login() throws IOException {

Scanner input = new Scanner(System.in);

String line;

BufferedReader fileReader = new BufferedReader(new FileReader("c:/temp/players.txt"));

//Split Player Information From file

while ((line = fileReader.readLine()) != null) {

String[] player = line.split(Pattern.quote(","));

this.allPlayers.add(new Player(player[0],player[2],player[1]));

}

System.out.println("Enter email : ");

String email = input.next();

//Check if player is already there or not

for (Player player : this.allPlayers)

{

if(player.Email.equals(email)) {

System.out.println("Enter pass : ");

String password = input.next();

if(player.Password.equals(password)) {

System.out.println("login");

return true;

}else {

System.out.println("wrong password ");

try {

this.login();

} catch (IOException e) {

// TODO Auto-generated catch block

e.printStackTrace();

}

}

}else {

System.out.println("wrong user ");

try {

this.login();

} catch (IOException e) {

// TODO Auto-generated catch block

e.printStackTrace();

}

}

}

return false;

}

}

### Class Topic

package p1;

import java.io.BufferedReader;

import java.io.BufferedWriter;

import java.io.FileReader;

import java.io.FileWriter;

import java.io.IOException;

import java.util.ArrayList;

import java.util.Scanner;

import java.util.regex.Pattern;

public class Topic {

String question;

ArrayList<String> options = new ArrayList<>();

String answer;

//Using array list to display Question.Option and answer

public Topic(String question, ArrayList<String> options, String answer) {

super();

this.question = question;

this.options = options;

this.answer = answer;

}

public Topic() {

}

//Getters and setters

public String getQuestion() {

return question;

}

public void setQuestion(String question) {

this.question = question;

}

public ArrayList<String> getOptions() {

return options;

}

public void setOptions(ArrayList<String> options) {

this.options = options;

}

public ArrayList<Topic> getQuestions(String type) throws IOException {

Scanner input = new Scanner(System.in);

ArrayList<Topic> allQuestions = new ArrayList<>();

BufferedReader fileReader;

String line;

if(type == "gk") {

fileReader = new BufferedReader(new FileReader("c:/temp/gk.txt"));

}else if(type == "sp") {

fileReader = new BufferedReader(new FileReader("c:/temp/sp.txt"));

}else{

fileReader = new BufferedReader(new FileReader("c:/temp/mov.txt"));

}

//Split Player Information From file using string and indexes

while ((line = fileReader.readLine()) != null) {

String[] gk = line.split(Pattern.quote(","));

String question;

ArrayList<String> options = new ArrayList<>();

String answer;

question = gk[0];

options.add(gk[1]);

options.add(gk[2]);

options.add(gk[3]);

options.add(gk[4]);

answer = gk[5];

allQuestions.add(new Topic(question,options,answer));

}

return allQuestions;

}

//creating temp string to show question and Options

public String toString() {

String tempString = "";

tempString += this.question + "\n";

int count = 1;

for(String option : this.options) {

tempString += count + ". " + option + "\n";

count++;

}

return tempString;

}

}

### Class User

package p1;

import java.io.IOException;

import java.util.ArrayList;

public class User {

//Properties For User

String Name;

String Password;

String Email;

ArrayList<Player> allPlayers = new ArrayList<>();

public User(String Name, String password, String email) {

this.Name = Name;

this.Password = password;

this.Email = email;

}

public User() {

}

//Getters and setters for User

public String getUserName() {

return this.Name;

}

public void setUserName(String userName) {

this.Name = userName;

}

public String getPassword() {

return this.Password;

}

public void setPassword(String password) {

this.Password = password;

}

public String getEmail() {

return this.Email;

}

public void setEmail(String email) {

this.Email = email;

}

public boolean login() throws IOException, IOException {

return false;

}

}