**Industry Management System**



Session: 2021 – 2024

**Submitted by:**

M Abdullah

2021-CS-104

**Supervised by:**

Mam Maida Shahid

**For Fulfillment of**

**OOP**

Department of Computer Science

**University of Engineering and Technology**

**Lahore Pakistan**

**Abstraction:**

**INDUSTRY Management System** is basically helpful for admin/manager in many different ways. It help in managing employees and also their work and pay. We can add or remove employee. We also made new admin or we can also change admin or delete admin. It is also helpful to check the stations or book the seat. All information is available on it. It is much helpful for us.

**Users of Application:**

­This is two user application.

1. **Admin**

* Admin can manage every feature of management system and can create login accounts for new admin and add staff information.

1. **User**

* User can be able to check the movies and the detail of the movie

For Example: he/she can check the release date of the movie or drama, he/she can also watch the director, producer, writer and actors;.

**Functional Requirements:**

1. **Admin**

* Edit Director.
* Edit Writer.
* Edit Producer.
* Edit Actor.
* Edit Employee.
* View Dramas.
* Exit

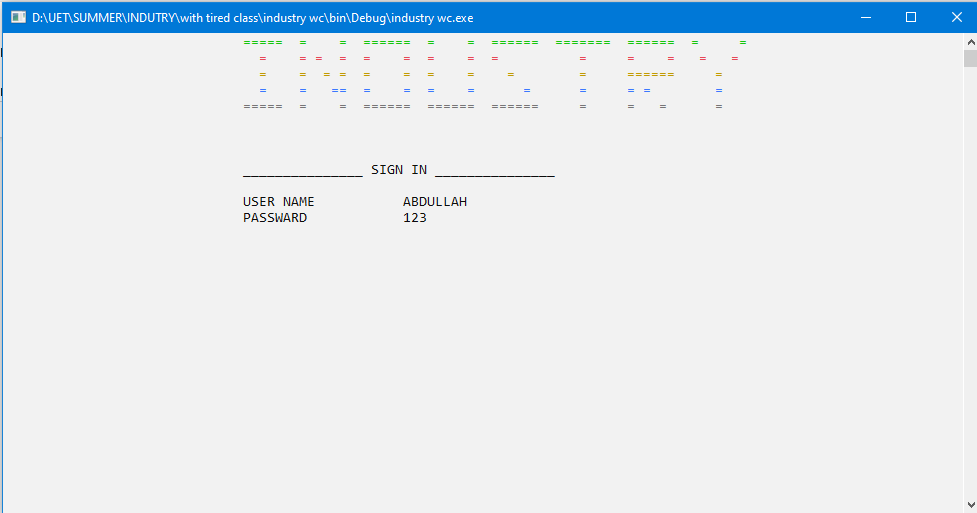
1. **Applicant**

* View Profile.
* View Pay.
* View Dramas.
* Exit

**Wireframes:**

* **Graphical user interface, application

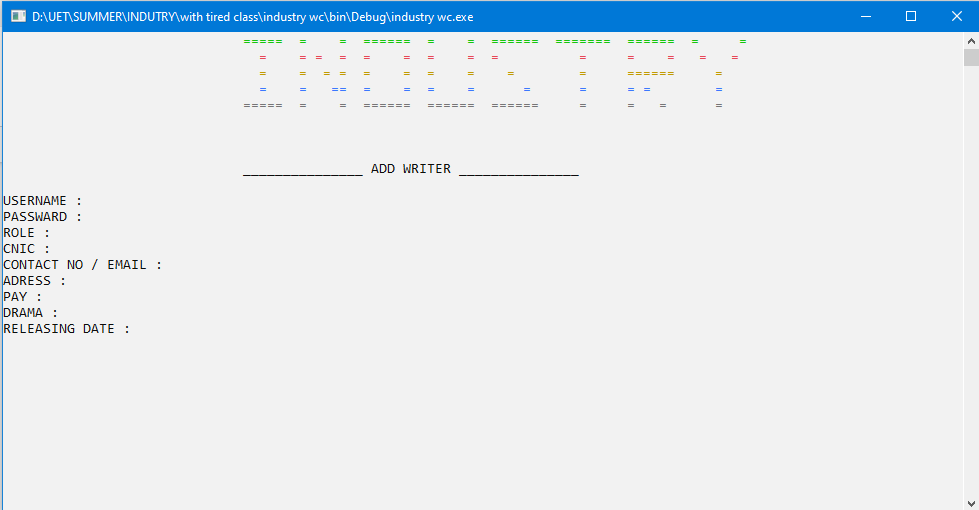
  Description automatically generatedMain page**
* **SIGN IN**



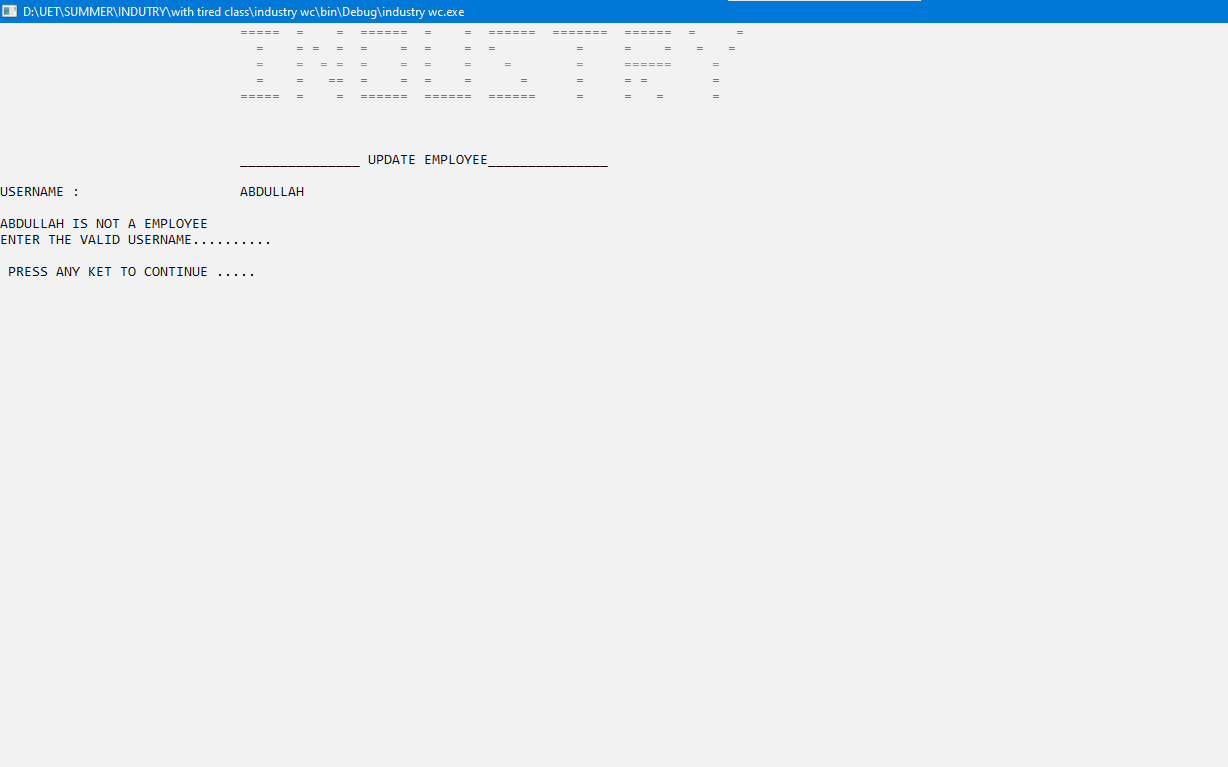
* **ADMIN**



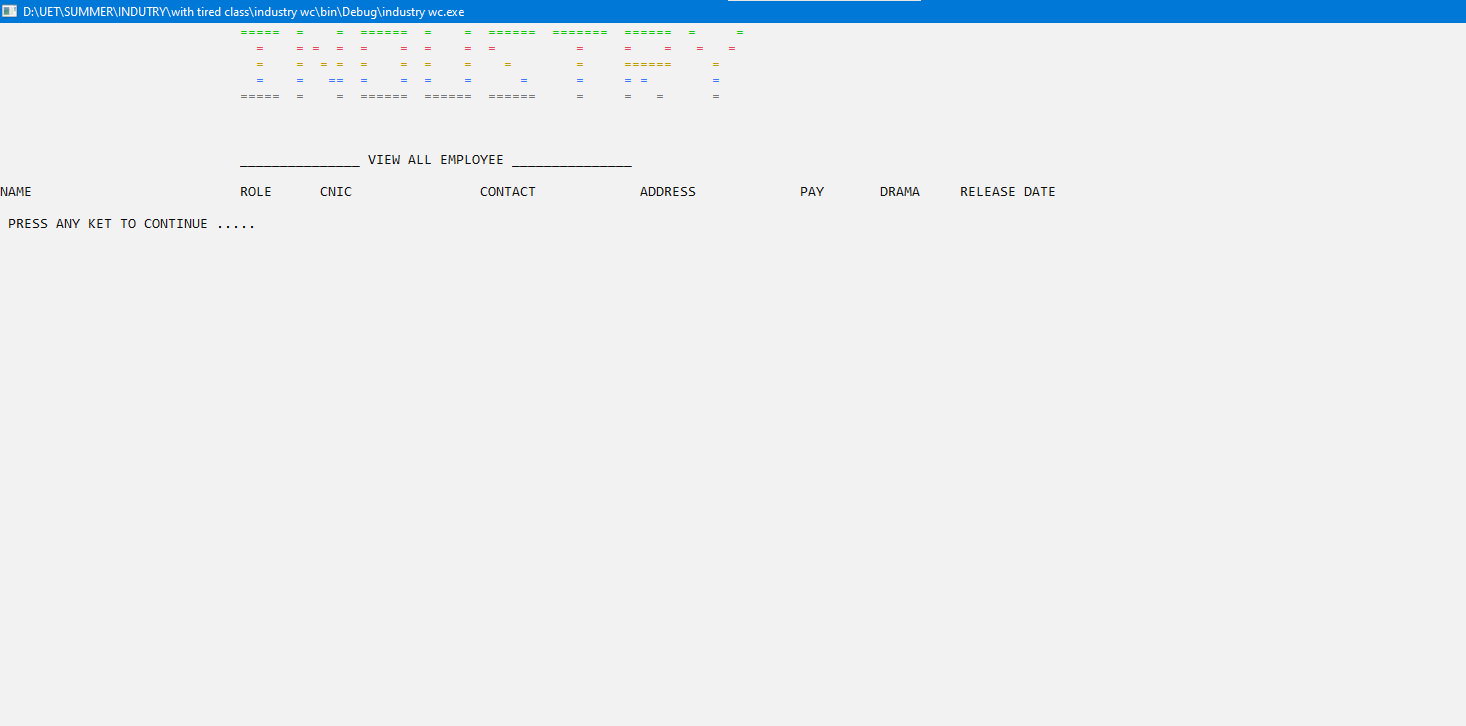
* **ADD CHARACTER**



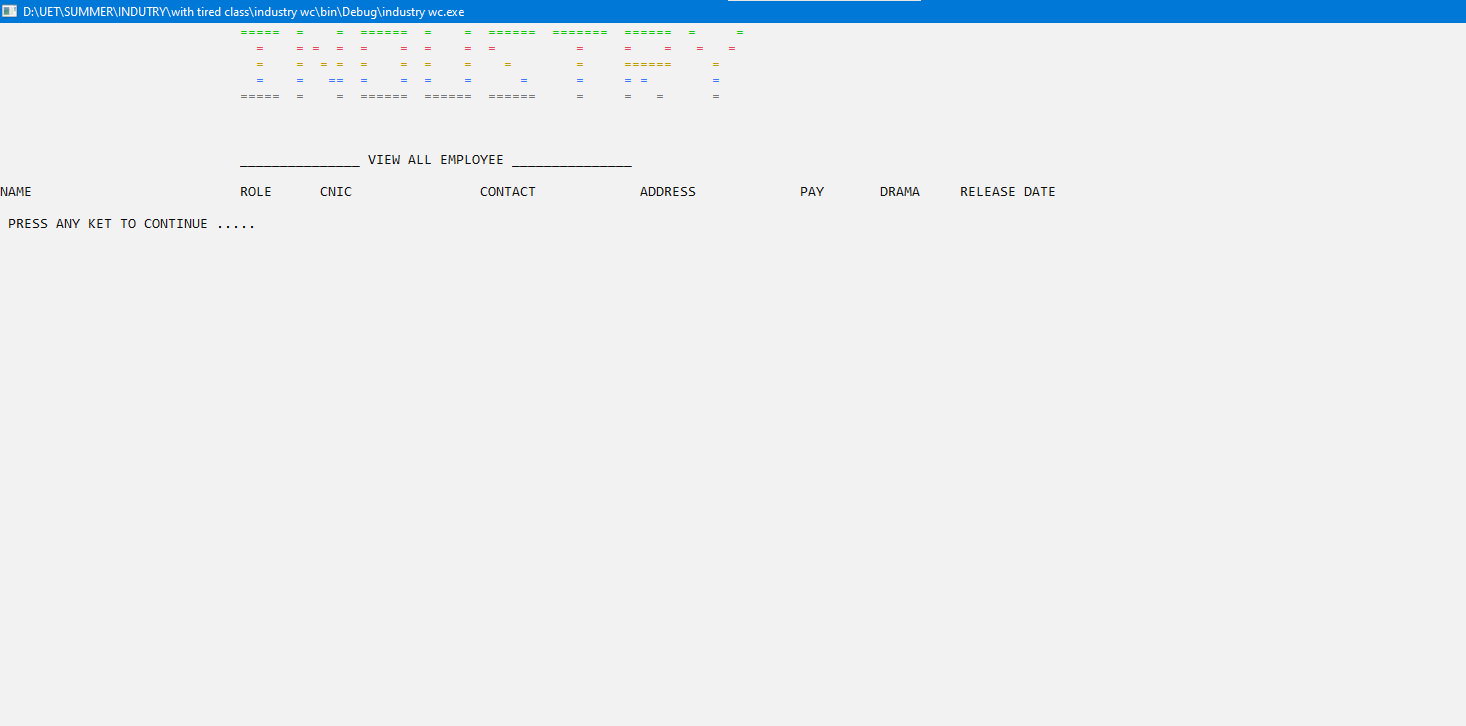
* **UPDATE EMPLOYEE INFO**



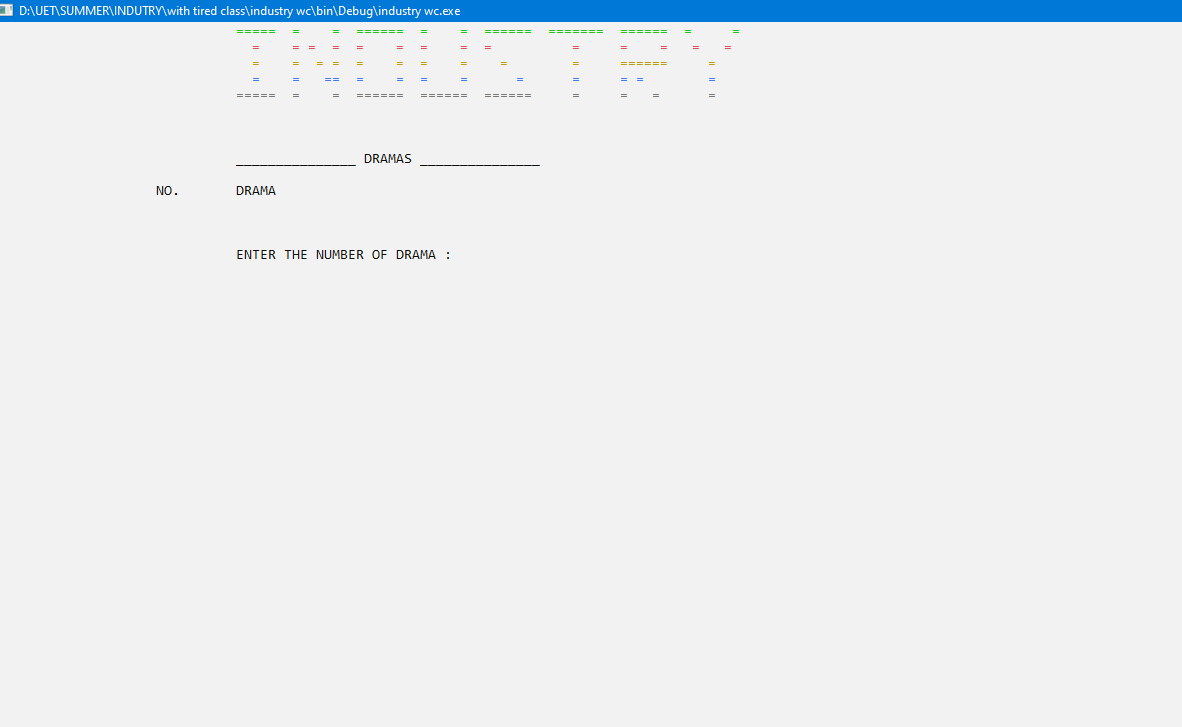
* **VIEW ALL EMPLOYEE**



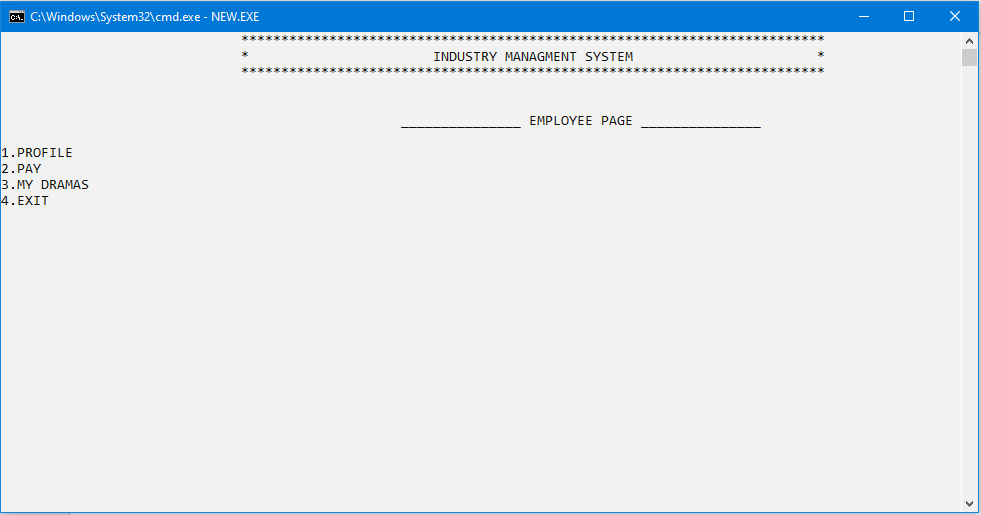
* **VIEW ALL**



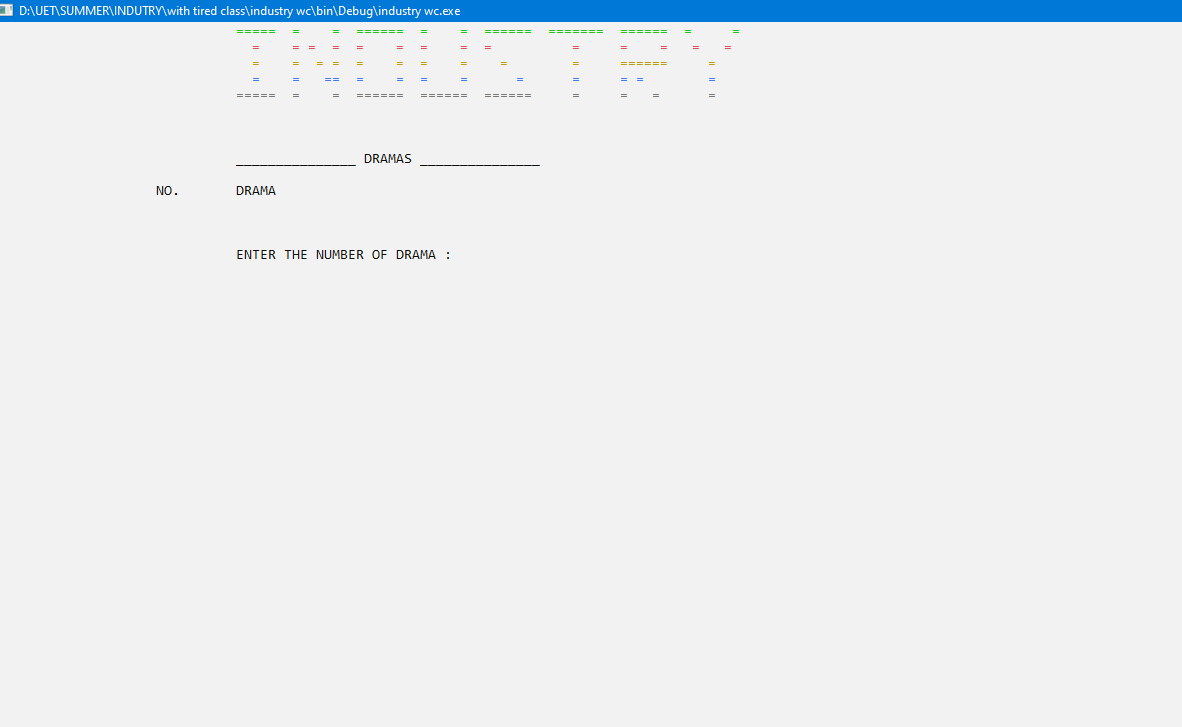
* **VIEW DRAMA**



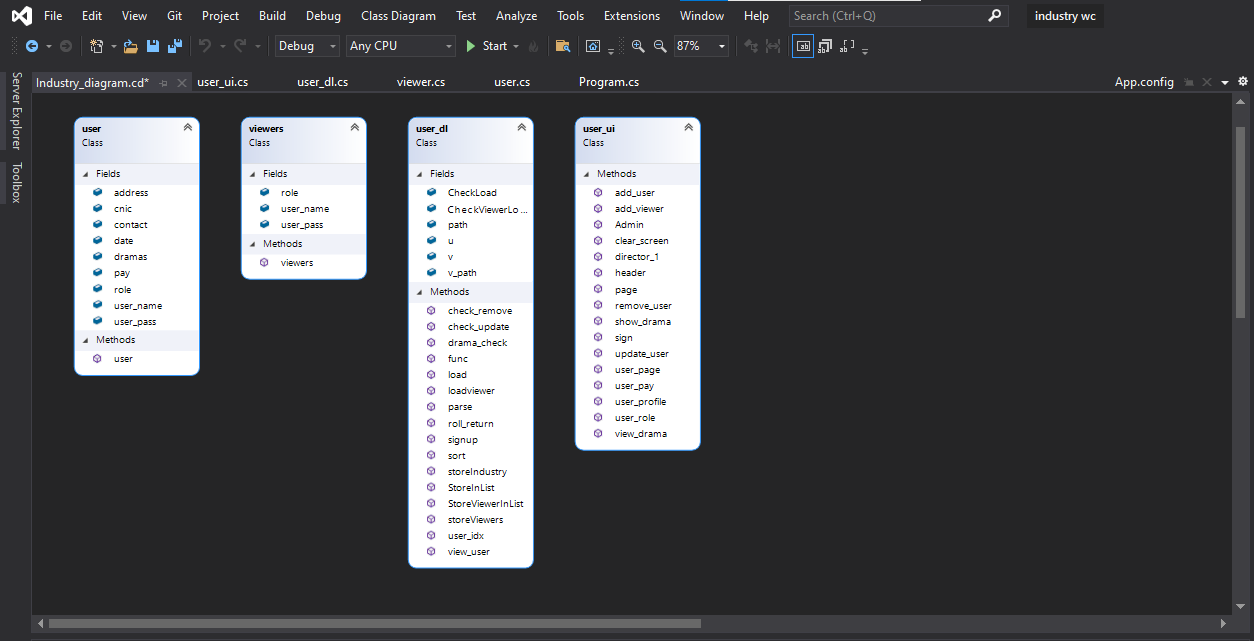
* **USER MENU**



* **VIEW DRAMAS**



**CLASS DIAGRAM:**



**MAIN:**

using System;

using System.Collections.Generic;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

using System.IO;

using industry\_wc.BL;

using industry\_wc.DL;

using industry\_wc.UI;

namespace industry\_wc

{

class Program

{

static void Main(string[] args)

{

string path = "industry.txt";

string v\_path = "viewers.txt";

const int min = 100;

string[] serial = new string[min], sdate = new string[min];

string main;

bool CheckLoad = user\_dl.load(path);

bool CheckViewerLoad = user\_dl.loadviewer(v\_path);

user\_dl.sort();

int dcount = user\_dl.drama\_check(serial, sdate);

String option = "1";

bool check = false;

while (check != true)

{

string rol;

option = user\_ui.sign();

if (option == "1")

{

main = user\_ui.page();

if (main == "DIRECTOR")

{

string op = "1";

while (op != "7")

{

op = user\_ui.Admin();

if (op == "1")

{

string op1 = "2";

while (op1 != "5")

{

rol = "DIRECTOR";

op1 = user\_ui.director\_1(rol);

if (op1 == "1")

{

user\_ui.add\_user( rol);

}

else if (op1 == "2")

{

user\_ui.remove\_user(rol);

}

else if (op1 == "3")

{

user\_ui.update\_user(rol);

}

else if (op1 == "4")

{

user\_dl.sort();

user\_dl.view\_user(rol);

}

else if (op1 != "5")

{

Console.WriteLine("ENTER THE VALID OPTION");

user\_ui.clear\_screen();

}

user\_dl.storeViewers(v\_path);

user\_dl.storeIndustry(path);

dcount = user\_dl.drama\_check( serial, sdate);

}

}

else if (op == "2")

{

string op1 = "1";

while (op1 != "5")

{

rol = "WRITER";

op1 = user\_ui.director\_1(rol);

if (op1 == "1")

{

user\_ui.add\_user( rol);

}

else if (op1 == "2")

{

user\_ui.remove\_user(rol);

}

else if (op1 == "3")

{

user\_ui.update\_user( rol);

}

else if (op1 == "4")

{

user\_dl.sort();

user\_dl.view\_user( rol);

}

else if (op1 != "5")

{

Console.WriteLine("ENTER THE VALID OPTION");

user\_ui.clear\_screen();

}

user\_dl.storeViewers(v\_path);

user\_dl.storeIndustry(path);

dcount = user\_dl.drama\_check(serial, sdate);

}

}

else if (op == "3")

{

string op1 = "2";

while (op1 != "5")

{

rol = "PRODUCER";

op1 = user\_ui.director\_1(rol);

if (op1 == "1")

{

user\_ui.add\_user( rol);

}

else if (op1 == "2")

{

user\_ui.remove\_user( rol);

}

else if (op1 == "3")

{

user\_ui.update\_user( rol);

}

else if (op1 == "4")

{

user\_dl.sort();

user\_dl.view\_user( rol);

}

else if (op1 != "5")

{

Console.WriteLine("ENTER THE VALID OPTION");

user\_ui.clear\_screen();

}

user\_dl.storeViewers(v\_path);

user\_dl.storeIndustry(path);

dcount = user\_dl.drama\_check( serial, sdate);

}

}

else if (op == "4")

{

string op1 = "2";

while (op1 != "5")

{

rol = "ACTOR";

op1 = user\_ui.director\_1(rol);

if (op1 == "1")

{

user\_ui.add\_user( rol);

}

else if (op1 == "2")

{

user\_ui.remove\_user( rol);

}

else if (op1 == "3")

{

user\_ui.update\_user( rol);

}

else if (op1 == "4")

{

user\_dl.sort();

user\_dl.view\_user( rol);

}

else if (op1 != "5")

{

Console.WriteLine("ENTER THE VALID OPTION");

user\_ui.clear\_screen();

}

user\_dl.storeViewers(v\_path);

user\_dl.storeIndustry(path);

dcount = user\_dl.drama\_check(serial, sdate);

}

}

else if (op == "5")

{

string op1 = "2";

while (op1 != "5")

{

rol = "EMPLOYEE";

op1 = user\_ui.director\_1(rol);

if (op1 == "1")

{

user\_ui.add\_user( rol);

}

else if (op1 == "2")

{

user\_ui.remove\_user(rol);

}

else if (op1 == "3")

{

user\_ui.update\_user( rol);

}

else if (op1 == "4")

{

user\_dl.sort();

user\_dl.view\_user( rol);

}

else if (op1 != "5")

{

Console.WriteLine("ENTER THE VALID OPTION");

user\_ui.clear\_screen();

}

user\_dl.storeViewers(v\_path);

user\_dl.storeIndustry(path);

dcount = user\_dl.drama\_check(serial, sdate);

}

}

else if (op == "6")

{

user\_dl.sort();

int opt = user\_ui.show\_drama(serial, sdate, dcount);

for (int x = 0; x < dcount; x++)

{

if (serial[opt - 1] == user\_dl.u[x].dramas && sdate[opt - 1] == user\_dl.u[x].date)

{

user\_ui.view\_drama(serial, sdate, x);

}

}

}

}

}

else if (main == "WRITER" || main == "PRODUCER" || main == "ACTOR" || main == "EMPLOYEE")

{

user\_dl.sort();

rol = main;

string op2 = "1";

while (op2 != "4")

{

op2 = user\_ui.user\_page(rol);

if (op2 == "1")

{

user\_ui.user\_profile(rol);

}

else if (op2 == "2")

{

user\_ui.user\_pay(rol );

}

else if (op2 == "3")

{

user\_ui.user\_role(rol);

}

else if (op2 != "4")

{

Console.WriteLine("ENTER THE VALID OPTION");

user\_ui.clear\_screen();

}

}

}

else if (main == "USER")

{

user\_dl.sort();

int opt = user\_ui.show\_drama(serial, sdate, dcount);

for (int x = 0; x < dcount; x++)

{

if (serial[opt - 1] == user\_dl.u[x].dramas && sdate[opt - 1] == user\_dl.u[x].date)

{

user\_ui.view\_drama(serial, sdate, x);

}

}

}

}

else if (option == "2")

{

rol = "user";

user\_dl.signup( );

user\_dl.storeViewers(v\_path);

user\_dl.storeIndustry(path);

}

else if (option == "3")

{

break;

}

}

}

}

}

**CLASSES**

class user

{

//==================================== > OBJECTS

public string user\_name;

public string user\_pass;

public string role;

public string cnic;

public string contact;

public string address;

public string dramas;

public string date;

public int pay;

//==================================== > CONSTRUCTORS

public user()

{

}

}

class viewers

{

public string user\_name;

public string user\_pass;

public string role;

public viewers()

{

}

}

class user\_dl

{

static public List<user> u = new List<user>();

static public List<viewers> v = new List<viewers>();

static public bool load(string path)

{

user u = new user();

if (File.Exists(path))

{

StreamReader file = new StreamReader(path);

string record;

while ((record = file.ReadLine()) != null)

{

u.user\_name = parse(record, 1);

u.user\_pass = parse(record, 2);

u.role = parse(record, 3);

u.cnic = parse(record, 4);

u.contact = parse(record, 5);

u.address = parse(record, 6);

u.pay = int.Parse(parse(record, 7));

u.dramas = parse(record, 8);

u.date = parse(record, 9);

StoreInList(u);

}

file.Close();

return true;

}

else

{

return false;

}

}

static public bool loadviewer(string v\_path)

{

if (File.Exists(v\_path))

{

StreamReader file = new StreamReader(v\_path);

string record;

while ((record = file.ReadLine()) != null)

{

viewers v = new viewers();

v.user\_name = parse(record, 1);

v.user\_pass = parse(record, 2);

v.role = parse(record, 3);

StoreViewerInList(v);

}

file.Close();

return true;

}

else

{

return false;

}

}

static public void sort()

{

int position = 0;

for (int i = 0; i < u.Count; i++)

{

position = i;

int large = func(position);

int temp = u[i].pay;

u[i].pay = u[large].pay;

u[large].pay = temp;

string temp\_1;

temp\_1 = u[i].user\_name;

u[i].user\_name = u[large].user\_name;

u[large].user\_name = temp\_1;

temp\_1 = u[i].role;

u[i].role = u[large].role;

u[large].role = temp\_1;

temp\_1 = u[i].cnic;

u[i].cnic = u[large].cnic;

u[large].cnic = temp\_1;

temp\_1 = u[i].contact;

u[i].contact = u[large].contact;

u[large].contact = temp\_1;

temp\_1 = u[i].address;

u[i].address = u[large].address;

u[large].address = temp\_1;

temp\_1 = u[i].dramas;

u[i].dramas = u[large].dramas;

u[large].dramas = temp\_1;

temp\_1 = u[i].date;

u[i].date = u[large].date;

u[large].date = temp\_1;

}

}

static public int func(int position)

{

int largest = u[position].pay;

int idx = position;

for (int i = position; i < u.Count; i++)

{

if (largest <= u[i].pay)

{

largest = u[i].pay;

idx = i;

}

}

return idx;

}

static public void StoreInList(user f)

{

u.Add(f);

}

static public void StoreViewerInList(viewers f)

{

v.Add(f);

}

static public string parse(string record, int field)

{

string term = "";

int comma = 1;

for (int i = 0; i < record.Length; i++)

{

if (record[i] == ',')

{

comma++;

}

else if (comma == field)

{

term = term + record[i];

}

}

return term;

}

static public void signup( )

{

string rol="USER";

user\_ui.add\_viewer(rol);

}

static public bool check\_user(string name, string pass)

{

int count = 0;

foreach(user i in u)

{

if(i.user\_name==name && i.user\_pass==pass)

{

count++;

}

}

foreach (viewers i in v)

{

if (i.user\_name == name && i.user\_pass == pass)

{

count++;

}

}

if (count>0)

{

return false;

}

return true;

}

static public string roll\_return(string user, string pass)

{

for (int x = 0; x < u.Count; x++)

{

if (user == u[x].user\_name && pass == u[x].user\_pass)

{

return u[x].role;

}

}

for (int x = 0; x < v.Count; x++)

{

if (user == v[x].user\_name && pass == v[x].user\_pass)

{

return v[x].role;

}

}

return "not include";

}

static public void storeIndustry(string path)

{

path = "industry.txt";

if (File.Exists(path))

{

StreamWriter file = new StreamWriter(path);

for (int i = 0; i < u.Count; i++)

{

file.WriteLine(u[i].user\_name + "," + u[i].user\_pass + "," + u[i].role + "," + u[i].cnic + "," + u[i].contact + "," + u[i].address + "," + u[i].pay + "," + u[i].dramas + "," + u[i].date);

}

file.Flush();

file.Close();

}

else

{

Console.WriteLine("file not exist");

Console.ReadKey();

}

}

static public void storeViewers(string v\_path)

{

if (File.Exists(v\_path))

{

StreamWriter file = new StreamWriter(v\_path);

for (int i = 0; i < v.Count; i++)

{

file.WriteLine(v[i].user\_name + "," + v[i].user\_pass + "," + v[i].role);

}

file.Flush();

file.Close();

}

else

{

Console.WriteLine("file not exist");

Console.ReadKey();

}

}

static public int user\_idx(string rol)

{

int idx = 0;

for (int x = 0; x < u.Count; x++)

{

if (rol == u[x].role)

{

return x;

}

}

return idx;

}

static public int drama\_check(string[] serial, string[] sdate)

{

int idx = 0;

for (int i = 0; i < u.Count; i++)

{

int count = 0;

for (int x = 0; x < serial.Length ; x++)

{

if (u[i].dramas == serial[x])

{

count++;

}

if (count > 0)

{

break;

}

}

if (count == 0)

{

serial[idx] = u[i].dramas;

sdate[idx] = u[i].date;

idx++;

}

}

Console.WriteLine(idx);

Console.ReadKey();

return idx;

}

static public bool check\_remove(string nam, string rol)

{

char y = ' ';

for (int x = 0; x < u.Count; x++)

{

if (nam == u[x].user\_name && u[x].role == rol)

{

Console.SetCursorPosition(15, 12);

Console.WriteLine("PRESS Y TO DELETE THIS USER ");

y = char.Parse(Console.ReadLine());

if (y == 'Y')

{

u.RemoveAt(x);

}

return true;

}

}

return false;

}

static public int check\_update(string nam, string rol)

{

for (int x = 0; x < u.Count; x++)

{

if (nam == u[x].user\_name && u[x].role == rol)

{

return x;

}

}

return 100000;

}

static public void view\_user(string rol)

{

user\_ui.header();

Console.SetCursorPosition(30, 8);

Console.WriteLine("\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ VIEW ALL " + rol + " \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_");

Console.SetCursorPosition(0, 10);

Console.WriteLine("NAME ");

Console.SetCursorPosition(30, 10);

Console.WriteLine("ROLE ");

Console.SetCursorPosition(40, 10);

Console.WriteLine("CNIC ");

Console.SetCursorPosition(60, 10);

Console.WriteLine("CONTACT ");

Console.SetCursorPosition(80, 10);

Console.WriteLine("ADDRESS ");

Console.SetCursorPosition(100, 10);

Console.WriteLine("PAY ");

Console.SetCursorPosition(110, 10);

Console.WriteLine("DRAMA ");

Console.SetCursorPosition(120, 10);

Console.WriteLine("RELEASE DATE ");

int y = 11;

for (int x = 0; x < u.Count; x++)

{

if (u[x].role == rol)

{

Console.SetCursorPosition(0, y);

Console.WriteLine(u[x].user\_name);

Console.SetCursorPosition(30, y);

Console.WriteLine(u[x].role);

Console.SetCursorPosition(40, y);

Console.WriteLine(u[x].cnic);

Console.SetCursorPosition(60, y);

Console.WriteLine(u[x].contact);

Console.SetCursorPosition(80, y);

Console.WriteLine(u[x].address);

Console.SetCursorPosition(100, y);

Console.WriteLine(u[x].pay);

Console.SetCursorPosition(110, y);

Console.WriteLine(u[x].dramas);

Console.SetCursorPosition(120, y);

Console.WriteLine(u[x].date);

y++;

}

}

user\_ui.clear\_screen();

}

}

class user\_ui

{

static public string page()

{

string name;

string pass;

bool flag = false;

while (flag == false)

{

header();

Console.SetCursorPosition(30, 8);

Console.WriteLine("\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ SIGN IN \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_");

Console.SetCursorPosition(30, 10);

Console.WriteLine("USER NAME ");

Console.SetCursorPosition(30, 11);

Console.WriteLine("PASSWARD");

Console.SetCursorPosition(50, 10);

name = Console.ReadLine();

Console.SetCursorPosition(50, 11);

pass = Console.ReadLine();

string check = user\_dl.roll\_return(name, pass);

if (check== "not include")

{

header();

Console.SetCursorPosition(40, 8);

Console.WriteLine("ENTER THE VALID USERNAME OR PASSWARD");

Console.WriteLine("");

clear\_screen();

}

else

{

flag = true;

return check;

}

}

return "1";

}

static public string Admin()

{

string op = "0";

header();

Console.SetCursorPosition(30, 8);

Console.WriteLine("\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ ADMIN MENU \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_");

Console.WriteLine("");

Console.WriteLine("1. DIRECTOR ");

Console.WriteLine("2. WRITER ");

Console.WriteLine("3. PRODUCER ");

Console.WriteLine("4. ACTOR ");

Console.WriteLine("5. EMPLOYEE ");

Console.WriteLine("6. DRAMAS ");

Console.WriteLine("7. EXIT ");

op = Console.ReadLine();

return op;

}

static public string director\_1(string rol)

{

header();

string op1 = "0";

Console.SetCursorPosition(30, 8);

Console.WriteLine("\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_" + rol + "\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_");

Console.WriteLine("");

Console.WriteLine("1. ADD ", rol);

Console.WriteLine("2. REMOVE ", rol);

Console.WriteLine("3. UPDATE ", rol);

Console.WriteLine("4. VIEW ALL ", rol);

Console.WriteLine("5. EXIT ");

op1 = Console.ReadLine();

return op1;

}

static public void add\_user(string rol)

{

user u = new user();

header();

Console.SetCursorPosition(30, 8);

Console.WriteLine("\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ ADD " + rol + " \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_");

Console.WriteLine("");

Console.SetCursorPosition(0, 10);

Console.WriteLine("USERNAME :");

Console.SetCursorPosition(0, 11);

Console.WriteLine("PASSWARD :");

Console.SetCursorPosition(0, 12);

Console.WriteLine("ROLE :");

u.role = rol;

Console.SetCursorPosition(0, 13);

Console.WriteLine("CNIC :");

Console.SetCursorPosition(0, 14);

Console.WriteLine("CONTACT NO / EMAIL :");

Console.SetCursorPosition(0, 15);

Console.WriteLine("ADRESS :");

Console.SetCursorPosition(0, 16);

Console.WriteLine("PAY :");

Console.SetCursorPosition(0, 17);

Console.WriteLine("DRAMA :");

Console.SetCursorPosition(0, 18);

Console.WriteLine("RELEASING DATE :");

Console.SetCursorPosition(30, 10);

u.user\_name = Console.ReadLine();

Console.SetCursorPosition(30, 11);

u.user\_pass = Console.ReadLine();

u.role = rol;

Console.SetCursorPosition(30, 12);

Console.WriteLine(u.role);

Console.SetCursorPosition(30, 13);

u.cnic = Console.ReadLine();

Console.SetCursorPosition(30, 14);

u.contact = Console.ReadLine();

Console.SetCursorPosition(30, 15);

u.address = Console.ReadLine();

Console.SetCursorPosition(30, 16);

u.pay = int.Parse(Console.ReadLine());

Console.SetCursorPosition(30, 17);

u.dramas = Console.ReadLine();

Console.SetCursorPosition(30, 18);

u.date = Console.ReadLine();

Console.SetCursorPosition(20, 24);

Console.WriteLine(u.role + " IS ADDED");

Console.WriteLine("");

user\_dl.StoreInList(u);

clear\_screen();

}

static public void add\_viewer(string rol)

{

viewers u = new viewers();

bool check =false;

while (check != true)

{

header();

Console.SetCursorPosition(30, 8);

Console.WriteLine("\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ ADD " + rol + " \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_");

Console.WriteLine("");

Console.SetCursorPosition(0, 10);

Console.WriteLine("USERNAME :");

Console.SetCursorPosition(0, 11);

Console.WriteLine("PASSWARD :");

Console.SetCursorPosition(0, 12);

Console.WriteLine("ROLE :");

u.role = rol;

string name, pass;

Console.SetCursorPosition(30, 10);

name = Console.ReadLine();

Console.SetCursorPosition(30, 11);

pass = Console.ReadLine();

if (user\_dl.check\_user(name,pass) == true)

{

u.user\_name = name;

u.user\_pass = pass;

check = true;

}

else

{

Console.SetCursorPosition(0, 14);

Console.WriteLine("THIS USER NAME ALREADY EXIST");

Console.SetCursorPosition(0, 15);

Console.WriteLine("USE ANOTHER USERNAME OR PASSWARD");

Console.WriteLine();

Console.SetCursorPosition(0, 17);

clear\_screen();

check =false;

}

}

u.role = rol;

Console.SetCursorPosition(30, 12);

Console.WriteLine(u.role);

Console.SetCursorPosition(20, 24);

Console.WriteLine(u.role + " IS ADDED");

Console.WriteLine("");

user\_dl.StoreViewerInList(u);

clear\_screen();

}

static public void remove\_user( string rol)

{

string nam;

header();

Console.SetCursorPosition(30, 8);

Console.WriteLine("\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ REMOVE " + rol + " \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_");

Console.WriteLine("");

Console.SetCursorPosition(0, 10);

Console.WriteLine("USERNAME :");

Console.SetCursorPosition(30, 10);

nam = Console.ReadLine();

bool check = user\_dl.check\_remove(nam, rol);

if(check==true)

{

Console.SetCursorPosition(0, 12);

Console.WriteLine(nam + " IS REMOVED.......");

}

else

{

Console.SetCursorPosition(0, 12);

Console.WriteLine(nam + " IS NOT A " + rol);

Console.WriteLine("");

Console.WriteLine("ENTER THE VALID USERNAME.......... ");

}

clear\_screen();

}

static public void update\_user(string rol)

{

string nam;

header();

Console.SetCursorPosition(30, 8);

Console.WriteLine("\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ UPDATE " + rol + "\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_");

Console.SetCursorPosition(0, 10);

Console.WriteLine("USERNAME :");

Console.SetCursorPosition(30, 10);

nam = Console.ReadLine();

char y = ' ';

int x = user\_dl.check\_update(nam, rol);

if (x != 100000)

{

header();

Console.SetCursorPosition(30, 8);

Console.WriteLine("\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ UPDATE ", rol, " \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_");

Console.SetCursorPosition(0, 10);

Console.WriteLine("UPDATE USERNAME :");

Console.SetCursorPosition(0, 11);

Console.WriteLine("UPDATE PASSWARD :");

Console.SetCursorPosition(0, 12);

Console.WriteLine("UPDATE ROLE :");

Console.SetCursorPosition(0, 13);

Console.WriteLine("UPDATE CNIC :");

Console.SetCursorPosition(0, 14);

Console.WriteLine("UPDATE CONTACT NO / EMAIL :");

Console.SetCursorPosition(0, 15);

Console.WriteLine("UPDATE ADRESS :");

Console.SetCursorPosition(0, 16);

Console.WriteLine("UPDATE PAY :");

Console.SetCursorPosition(0, 17);

Console.WriteLine("UPDATE DRAMA :");

Console.SetCursorPosition(0, 18);

Console.WriteLine("UPDATE RELEASING DATE :");

Console.SetCursorPosition(30, 10);

user\_dl.u[x].user\_name = Console.ReadLine();

Console.SetCursorPosition(30, 11);

user\_dl.u[x].user\_pass = Console.ReadLine();

Console.SetCursorPosition(30, 12);

user\_dl.u[x].role = Console.ReadLine();

Console.SetCursorPosition(30, 13);

user\_dl.u[x].cnic = Console.ReadLine();

Console.SetCursorPosition(30, 14);

user\_dl.u[x].contact = Console.ReadLine();

Console.SetCursorPosition(30, 15);

user\_dl.u[x].address = Console.ReadLine();

Console.SetCursorPosition(30, 16);

user\_dl.u[x].pay = int.Parse(Console.ReadLine());

Console.SetCursorPosition(30, 17);

user\_dl.u[x].dramas = Console.ReadLine();

Console.SetCursorPosition(30, 18);

user\_dl.u[x].date = Console.ReadLine();

y = 'Y';

}

else

{

Console.SetCursorPosition(0, 12);

Console.WriteLine(nam + " IS NOT A " + rol);

Console.WriteLine("ENTER THE VALID USERNAME.......... ");

}

if (y == 'y')

{

Console.SetCursorPosition(0, 12);

Console.WriteLine(nam + " IS UPDATED.......");

}

clear\_screen();

}

static public void header()

{

Console.Clear();

ConsoleColor color = ConsoleColor.Green;

Console.ForegroundColor = color;

Console.SetCursorPosition(30, 0);

Console.WriteLine("===== = = ====== = = ====== ======= ====== = =");

color = ConsoleColor.Red;

Console.ForegroundColor = color;

Console.SetCursorPosition(30, 1);

Console.WriteLine(" = = = = = = = = = = = = = =");

color = ConsoleColor.DarkYellow;

Console.ForegroundColor = color;

Console.SetCursorPosition(30, 2);

Console.WriteLine(" = = = = = = = = = = ====== =");

color = ConsoleColor.Blue;

Console.ForegroundColor = color;

Console.SetCursorPosition(30, 3);

Console.WriteLine(" = = == = = = = = = = = =");

color = ConsoleColor.DarkGray;

Console.ForegroundColor = color;

Console.SetCursorPosition(30, 4);

Console.WriteLine("===== = = ====== ====== ====== = = = =");

color = ConsoleColor.Black;

Console.ForegroundColor = color;

}

static public void clear\_screen()

{

Console.WriteLine("");

Console.WriteLine(" PRESS ANY KET TO CONTINUE ..... ");

Console.ReadLine();

Console.Clear();

}

static public string sign()

{

string op = "0";

while (op != "3")

{

header();

Console.SetCursorPosition(30, 8);

Console.WriteLine("\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ SIGN UP / SIGN IN\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_");

Console.SetCursorPosition(30, 10);

Console.WriteLine("1.SIGN IN ");

Console.SetCursorPosition(30, 11);

Console.WriteLine("2 SIGN UP ");

Console.SetCursorPosition(30, 12);

Console.WriteLine("3.EXIT ");

Console.WriteLine("");

Console.SetCursorPosition(30, 13);

Console.WriteLine("ENTER THE OPTION");

Console.SetCursorPosition(60, 13);

op = Console.ReadLine();

if (op != "1" && op != "2" && op != "3")

{

header();

Console.SetCursorPosition(30, 8);

Console.WriteLine("\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ SIGN UP / SIGN IN\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_");

Console.SetCursorPosition(30, 10);

Console.WriteLine("ENTER THE RIGHT OPTION");

Console.SetCursorPosition(30, 11);

Console.WriteLine("PRESS ANY KEY TO CONTINUE");

Console.ReadKey();

}

else

{

return op;

}

}

return "3";

}

static public void user\_role(string rol)

{

header();

Console.SetCursorPosition(30, 8);

int y = 10;

Console.WriteLine("\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ " + rol + " ROLE \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_");

Console.SetCursorPosition(0, y);

Console.WriteLine("NAME ");

Console.SetCursorPosition(20, y);

Console.WriteLine("ROLE ");

Console.SetCursorPosition(40, y);

Console.WriteLine("DRAMA ");

Console.SetCursorPosition(60, y);

int n = 12;

int x = user\_dl.user\_idx(rol);

Console.SetCursorPosition(0, n);

Console.WriteLine(user\_dl.u[x].user\_name);

Console.SetCursorPosition(20, n);

Console.WriteLine(user\_dl.u[x].role);

Console.SetCursorPosition(40, n);

Console.WriteLine(user\_dl.u[x].dramas);

Console.SetCursorPosition(60, n);

clear\_screen();

}

static public void user\_pay(string rol)

{

header();

Console.SetCursorPosition(30, 8);

Console.WriteLine("\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ " + rol + " PAY \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_");

Console.WriteLine("");

Console.SetCursorPosition(0, 10);

Console.WriteLine("NAME ");

Console.SetCursorPosition(20, 10);

Console.WriteLine("ROLE ");

Console.SetCursorPosition(40, 10);

Console.WriteLine("PAY ");

Console.SetCursorPosition(60, 10);

int x = user\_dl.user\_idx(rol);

Console.SetCursorPosition(0, 12);

Console.WriteLine(user\_dl.u[x].user\_name);

Console.SetCursorPosition(20, 12);

Console.WriteLine(user\_dl.u[x].role);

Console.SetCursorPosition(40, 12);

Console.WriteLine(user\_dl.u[x].pay);

Console.SetCursorPosition(60, 12);

clear\_screen();

}

static public void user\_profile(string rol )

{

header();

Console.SetCursorPosition(30, 8);

Console.WriteLine("\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ " + rol + " PROFILE \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_");

Console.WriteLine("");

Console.SetCursorPosition(0, 10);

Console.WriteLine("NAME ");

Console.SetCursorPosition(20, 10);

Console.WriteLine("PASSWARD ");

Console.SetCursorPosition(30, 10);

Console.WriteLine("ROLE ");

Console.SetCursorPosition(40, 10);

Console.WriteLine("CNIC ");

Console.SetCursorPosition(60, 10);

Console.WriteLine("CONTACT ");

Console.SetCursorPosition(80, 10);

Console.WriteLine("ADDRESS ");

Console.SetCursorPosition(100, 10);

Console.WriteLine("PAY ");

Console.SetCursorPosition(110, 10);

Console.WriteLine("DRAMA ");

Console.SetCursorPosition(120, 10);

Console.WriteLine("RELEASE DATE ");

int y = 12;

int x = user\_dl.user\_idx(rol);

Console.SetCursorPosition(0, y);

Console.WriteLine(user\_dl.u[x].user\_name);

Console.SetCursorPosition(20, y);

Console.WriteLine(user\_dl.u[x].user\_pass);

Console.SetCursorPosition(30, y);

Console.WriteLine(user\_dl.u[x].role);

Console.SetCursorPosition(40, y);

Console.WriteLine(user\_dl.u[x].cnic);

Console.SetCursorPosition(60, y);

Console.WriteLine(user\_dl.u[x].contact);

Console.SetCursorPosition(80, y);

Console.WriteLine(user\_dl.u[x].address);

Console.SetCursorPosition(100, y);

Console.WriteLine(user\_dl.u[x].pay);

Console.SetCursorPosition(110, y);

Console.WriteLine(user\_dl.u[x].dramas);

Console.SetCursorPosition(120, y);

Console.WriteLine(user\_dl.u[x].date);

y++;

clear\_screen();

}

static public void view\_drama(string[] serial, string[] sdate, int idx)

{

header();

Console.SetCursorPosition(30, 8);

Console.WriteLine("\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ DRAMA DETAILS\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_");

Console.WriteLine("");

Console.SetCursorPosition(30, 10);

Console.WriteLine("DRAMA :");

Console.SetCursorPosition(30, 11);

Console.WriteLine("RELEASE DATE :");

for (int i = 0; i < user\_dl.u.Count; i++)

{

Console.SetCursorPosition(60, 10);

Console.WriteLine(serial[idx]);

Console.SetCursorPosition(60, 11);

Console.WriteLine(sdate[idx]);

int d = 0, p = 0, w = 0, a = 0;

for (int x = 0; x < user\_dl.u.Count; x++)

{

if (user\_dl.u[x].dramas == serial[idx] && user\_dl.u[x].date == sdate[idx])

{

Console.SetCursorPosition(35, 13);

Console.WriteLine("DIRECTOR");

if (user\_dl.u[x].role == "DIRECTOR")

{

Console.SetCursorPosition(45, 13);

Console.WriteLine(user\_dl.u[x].user\_name);

d++;

}

Console.SetCursorPosition(35, 14);

Console.WriteLine("WRITER");

if (user\_dl.u[x].role == "WRITER")

{

Console.SetCursorPosition(45, 14);

Console.WriteLine(user\_dl.u[x].user\_name);

w++;

}

Console.SetCursorPosition(35, 15);

Console.WriteLine("PRODUCER");

if (user\_dl.u[x].role == "PRODUCER")

{

Console.SetCursorPosition(45, 15);

Console.WriteLine(user\_dl.u[x].user\_name);

p++;

}

Console.SetCursorPosition(35, 16);

Console.WriteLine("ACTOR");

if (user\_dl.u[x].role == "ACTOR")

{

Console.SetCursorPosition(45, 16);

Console.WriteLine(user\_dl.u[x].user\_name);

a++;

}

}

}

if (d == 0)

{

Console.SetCursorPosition(45, 13);

Console.WriteLine("NEED DIRECTOR");

}

if (w == 0)

{

Console.SetCursorPosition(45, 14);

Console.WriteLine("NEED WRITER");

}

if (p == 0)

{

Console.SetCursorPosition(45, 15);

Console.WriteLine("NEED PRODUCER");

}

if (a == 0)

{

Console.SetCursorPosition(45, 16);

Console.WriteLine("NEED ACTORS");

}

}

clear\_screen();

}

static public string user\_page(string rol)

{

header();

string op = "0";

Console.SetCursorPosition(30, 8);

Console.WriteLine("\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ " + rol + " PAGE \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_");

Console.SetCursorPosition(0, 10);

Console.WriteLine("1.PROFILE ");

Console.SetCursorPosition(0, 11);

Console.WriteLine("2.PAY");

Console.SetCursorPosition(0, 12);

Console.WriteLine("3.MY DRAMAS");

Console.SetCursorPosition(0, 13);

Console.WriteLine("4.EXIT");

Console.SetCursorPosition(0, 14);

op = Console.ReadLine();

return op;

}

static public int show\_drama(string[] serial, string[] sdate, int dcount)

{

header();

Console.SetCursorPosition(30, 8);

Console.WriteLine("\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ DRAMAS \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_");

Console.WriteLine("");

Console.SetCursorPosition(20, 10);

Console.WriteLine("NO.");

Console.SetCursorPosition(30, 10);

Console.WriteLine("DRAMA ");

int y = 12;

int x = 1;

for (int i = 0; i < dcount; i++)

{

Console.SetCursorPosition(20, y);

Console.WriteLine(x);

Console.SetCursorPosition(30, y);

Console.WriteLine(serial[i]);

y++;

x++;

}

int point;

y++;

y++;

Console.SetCursorPosition(30, y);

Console.WriteLine("ENTER THE NUMBER OF DRAMA : ");

Console.SetCursorPosition(60, y);

point = int.Parse(Console.ReadLine());

return point;

}

}