Name: Aryan Patel

SIN: 301226774

Drive Class

using System;

using System.Collections.Generic;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

namespace Assignment3

{

[Flags]

public enum Genre

{

Unrated,

Action,

Adventure,

Animation,

Comedy,

Documentary,

Fantasy,

Horror,

Musical,

Mystery,

Romance,

}

public enum Day

{

Sun,

Mon,

Tue,

Wed,

Thu,

Fri,

Sat,

}

class Program

{

static void Main(string[] args)

{

Movie terminator = new Movie("Terminator 2: Judgement Day", 1991, 105);

terminator.AddActor("Arnold Schwarzenegger");

terminator.SetGenre(Genre.Horror | Genre.Action);

terminator.AddActor("Linda Hamilton");

Show s1 = new Show(terminator, Day.Mon, 5.95, new Time(11, 35));

Console.WriteLine(s1); //displays one object

Theatre eglinton = new Theatre("Cineplex");

eglinton.AddShow(s1);

eglinton.PrintShows(); //displays one object

Movie godzilla = new Movie("Godzilla 2014", 2014, 123);

godzilla.AddActor("Aaron Johnson");

godzilla.AddActor("Ken Watanabe");

godzilla.AddActor("Elizabeth Olsen");

godzilla.SetGenre(Genre.Action | Genre.Documentary | Genre.Comedy);

Movie trancendence = new Movie("Transendence", 2014, 120);

trancendence.AddActor("Johnny Depp");

trancendence.AddActor("Morgan Freeman");

trancendence.SetGenre(Genre.Comedy);

eglinton.AddShow(new Show(trancendence, Day.Sun, 7.87, new Time(18, 5)));

Movie m1 = new Movie("The Shawshank Redemption", 1994, 120);

m1.AddActor("Tim Robbins");

m1.AddActor("Morgan Freeman");

m1.SetGenre(Genre.Action);

eglinton.AddShow(new Show(m1, Day.Sun, 8.87, new Time(14, 5)));

Movie avengers = new Movie("Avengers: Endgame", 2019, 120);

avengers.AddActor("Robert Downey Jr.");

avengers.AddActor("Chris Evans");

avengers.AddActor("Chris Hemsworth");

avengers.AddActor("Scarlett Johansson");

avengers.AddActor("Mark Ruffalo");

avengers.SetGenre(Genre.Action | Genre.Fantasy | Genre.Adventure);

eglinton.AddShow(new Show(avengers, Day.Sat, 12.25, new Time(21, 5)));

m1 = new Movie("Olympus Has Fallen", 2013, 120);

m1.AddActor("Gerard Butler");

m1.AddActor("Morgan Freeman");

m1.SetGenre(Genre.Action);

eglinton.AddShow(new Show(m1, Day.Sun, 8.87, new Time(16, 5)));

m1 = new Movie("The Mask of Zorro", 1998, 136);

m1.AddActor("Antonio Banderas");

m1.AddActor("Anthony Hopkins");

m1.AddActor("Catherine Zeta-Jones");

m1.SetGenre(Genre.Action | Genre.Romance);

eglinton.AddShow(new Show(m1, Day.Sun, 8.87, new Time(16, 5)));

m1 = new Movie("Four Weddings and a Funeral", 1994, 117);

m1.AddActor("Hugh Grant");

m1.AddActor("Andie MacDowell");

m1.AddActor("Kristin Scott Thomas");

m1.SetGenre(Genre.Comedy | Genre.Romance);

eglinton.AddShow(new Show(m1, Day.Sat, 8.87, new Time(15, 5)));

eglinton.AddShow(new Show(m1, Day.Tue, 6.50, new Time(16, 5)));

m1 = new Movie("You've Got Mail", 1998, 119);

m1.AddActor("Tom Hanks");

m1.AddActor("Meg Ryan");

m1.SetGenre(Genre.Comedy | Genre.Romance);

eglinton.AddShow(new Show(m1, Day.Sat, 8.87, new Time(15, 5)));

m1 = new Movie("The Poison Rose", 2019, 98);

m1.AddActor("John Travolta");

m1.AddActor("Morgan Freeman");

m1.AddActor("Brendan Fraser");

m1.SetGenre(Genre.Action | Genre.Romance);

eglinton.AddShow(new Show(m1, Day.Sun, 10.25, new Time(22, 5)));

Movie car3 = new Movie("Cars 3", 2017, 109);

car3.AddActor("Owen Williams");

car3.AddActor("Cristela Alonzo");

car3.AddActor("Arnie Hammer");

car3.AddActor("Chris Cooper");

car3.SetGenre(Genre.Comedy | Genre.Animation | Genre.Romance);

eglinton.AddShow(new Show(car3, Day.Sat, 6.40, new Time(09, 55)));

eglinton.AddShow(new Show(car3, Day.Sat, 6.50, new Time(11, 05)));

Movie toys4 = new Movie("Toys Story 4", 2019, 89);

toys4.AddActor("Keanu Reeves");

toys4.AddActor("Christina Hendricks");

toys4.AddActor("Tom Hanks");

toys4.AddActor("Tim Allen");

toys4.SetGenre(Genre.Comedy | Genre.Fantasy | Genre.Animation);

eglinton.AddShow(new Show(toys4, Day.Sat, 6.40, new Time(14, 10)));

eglinton.AddShow(new Show(godzilla, Day.Mon, 6.89, new Time(13, 55)));

eglinton.AddShow(new Show(avengers, Day.Sat, 12.25, new Time(21, 5)));

eglinton.AddShow(new Show(godzilla, Day.Sun, 6.89, new Time(14, 0)));

eglinton.AddShow(new Show(toys4, Day.Sat, 6.40, new Time(14, 10)));

eglinton.AddShow(new Show(avengers, Day.Sat, 12.25, new Time(21, 5)));

eglinton.AddShow(new Show(godzilla, Day.Sun, 6.89, new Time(16, 55)));

eglinton.AddShow(new Show(avengers, Day.Sat, 12.25, new Time(21, 5)));

eglinton.AddShow(new Show(m1, Day.Sat, 10.25, new Time(20, 35)));

eglinton.AddShow(new Show(godzilla, Day.Wed, 8.50, new Time(22, 5)));

eglinton.AddShow(new Show(avengers, Day.Tue, 10.75, new Time(20, 30)));

eglinton.AddShow(new Show(godzilla, Day.Thu, 8.50, new Time(20, 15)));

eglinton.AddShow(new Show(avengers, Day.Wed, 10.75, new Time(20, 30)));

eglinton.AddShow(new Show(godzilla, Day.Fri, 8.50, new Time(18, 25)));

eglinton.AddShow(new Show(avengers, Day.Sun, 10.75, new Time(14, 15)));

eglinton.PrintShows(); //displays 27 objects

eglinton.PrintShows(Day.Sun); //displays 8 objects

eglinton.PrintShows(Genre.Action); //displays 19 objects

eglinton.PrintShows(Genre.Romance); //displays 8 objects

eglinton.PrintShows(Genre.Action | Genre.Romance); //displays 3 objects

eglinton.PrintShows("Morgan Freeman"); //displays 5 objects

Time time = new Time(14, 05);

eglinton.PrintShows(time); //displays 6 objects

eglinton.PrintShows(Day.Sun, time); //displays 3 objects

}

}

}

Movie class

using System;

using System.Collections.Generic;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

namespace Assignment3

{

class Movie

{

public int Length { get; }

public int Year { get; }

public string Title { get; }

public Genre Genre { get; private set; }

public List<string> Cast { get; }

public Movie(string name, int year, int length)

{

this.Title = name;

this.Year = year;

this.Length = length;

this.Cast = new List<string>();

}

public void AddActor(string actor)

{

this.Cast.Add(actor);

}

public void SetGenre(Genre genre)

{

this.Genre = genre;

}

public override string ToString()

{

return $"\nTitle:- {this.Title} ({this.Year.ToString()})\n" +

$"\nLength:- {this.Length.ToString()} min\n" +

$"\nGenre:- {this.Genre.ToString()}\n" +

$"\nCast:- { String.Join(", ", Cast)}\n";

}

}

}

Time Class

using System;

using System.Collections.Generic;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

namespace Assignment3

{

internal class Time

{

public int Hours { get; }

public int Minutes { get; }

public Time(int hours,int minutes=0)

{

this.Hours = hours;

this.Minutes = minutes;

}

public override string ToString()

{

return $"{this.Hours.ToString()}:{this.Minutes.ToString()}";

}

public static bool operator ==(Time lhs,Time rhs)

{

if ((lhs.Minutes + lhs.Hours \* 60) == (rhs.Hours \* 60 + rhs.Minutes))

{

return true;

}

else return false;

}

public static bool operator !=(Time lhs, Time rhs)

{

if ((lhs.Minutes + lhs.Hours \* 60) == (rhs.Hours \* 60 + rhs.Minutes))

{

return false;

}

else return true;

}

}

}

Show class

using System;

using System.Collections.Generic;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

namespace Assignment3

{

class Show

{

public double Price { get; }

public Day Day { get; }

public Movie Movie { get; }

public Time Time { get; }

public Show(Movie movie, Day day, double price, Time time)

{

this.Price = price;

this.Day = day;

this.Movie = movie;

this.Time = time;

}

public override string ToString()

{

return $"\n---Movie Details----\n{this.Movie.ToString()} \n" +

$"Release Date:- {this.Day.ToString()} \n" +

$"\nTime of the show:- {this.Time.ToString()}\n" +

$"\nPrice of the show:- ${this.Price.ToString()}\n";

}

}

}

Theatre class

using System;

using System.Collections.Generic;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

namespace Assignment3

{

class Theatre

{

private List<Show> show;

public string Name { get; }

public Theatre(string name)

{

this.Name = name;

this.show = new List<Show>();

}

public void AddShow(Show show)

{

this.show.Add(show);

}

public void PrintShows()

{

foreach(var s in show)

{

Console.WriteLine(s.ToString());

}

}

public void PrintShows(Genre genre)

{

foreach (var s in show)

{

if(s.Movie.Genre.HasFlag(genre))

Console.WriteLine(s.ToString());

}

}

public void PrintShows(Day day)

{

foreach (var s in show)

{

if(s.Day == day)

Console.WriteLine(s.ToString());

}

}

public void PrintShows(Time time)

{

foreach (var s in show)

{

if (s.Time == time)

Console.WriteLine(s.ToString());

}

}

public void PrintShows(string actor)

{

foreach (var s in show)

{

if(s.Movie.Cast.Contains(actor))

Console.WriteLine(s.ToString());

}

}

public void PrintShows(Day day, Time time)

{

foreach (var s in show)

{

if(s.Day == day && s.Time == time)

Console.WriteLine(s.ToString());

}

}

}

}