**Final – Daniele Ricciardelli**

**Screenshot**

**A screenshot of a computer

Description automatically generated**

**A screenshot of a computer program

Description automatically generated with medium confidence**

**Code (main)**

using Final\_Exam;

using System;

using System.Collections.Generic;

using System.Diagnostics;

using System.Linq;

using System.Runtime.Remoting.Metadata.W3cXsd2001;

using System.Text;

using System.Threading.Tasks;

namespace Final\_Exam

{

internal class Program

{

static void Main(string[] args)

{

Console.WriteLine("Requested program with their outputs:\n");

Date dateOfPublicationMyBook = new Date { Month = 05, Day = 16, Year = 2023 };

Book myBook = new Book("Jhon Smith", dateOfPublicationMyBook, "C++ Programming", 100.00 );

Console.WriteLine($"Title: {myBook.tittle}\t\t Author: {myBook.author}\t\t\t Price: ${myBook.price}\n");

Magazine hermag = new Magazine ("Paula Wang", "Keto for Life", 20.00 );

Console.WriteLine($"Title: {hermag.tittle}\t\t Editor-in-Chief: {hermag.editorInChief}\t\t Price: ${hermag.price}\n\n");

Console.WriteLine("----------------------------------------\n");

hermag.Discount(1.20); // Through a method-function

Console.WriteLine($"Title: {hermag.tittle}, Editor-in-Chief: {hermag.editorInChief}, Price: ${hermag.price}\n");

hermag.price = 19.80; // Through mutator

Console.WriteLine($"Title: {hermag.tittle}, Editor-in-Chief: {hermag.editorInChief}, Price: ${hermag.price}\n");

Console.WriteLine($"Year of Publication: {myBook.dateOfPublication.Year}\n");

Console.WriteLine("----------------------------------------");

Console.WriteLine("\nExtra functions and code\n");

Console.WriteLine("Would you like to donate your book? Y/N");

string answer1;

answer1 = Console.ReadLine();

if (answer1 == "Y")

{

myBook.Donate();

}

else if (answer1 == "N")

{

myBook.Read();

}

else

{

Console.WriteLine("Wrong Input");

}

Console.WriteLine("");

Console.WriteLine("I noticed you liked the magazine. Would you like to subscribe? Y/N");

string answer2;

answer2 = Console.ReadLine();

if (answer2 == "Y")

{

hermag.Subscribe();

}

else if (answer2 == "N")

{

Console.WriteLine("My mistake.");

}

else

{

Console.WriteLine("Wrong Input");

}

Console.WriteLine("");

}

}

}

**Code (Date)**

using System;

using System.Collections.Generic;

using System.Linq;

using System.Security.Policy;

using System.Text;

using System.Threading.Tasks;

namespace Final\_Exam

{

public class Date

{

// Data Members

public int month;

public int day;

public int year;

// Default C.

public Date()

{

month = 0;

day = 0;

year = 0;

}

// Overload C.

public Date(int mm, int dd, int yy)

{

month = mm;

day = dd;

year = yy;

}

// Get n Sets

public int Month

{

get { return month; }

set { month = value; }

}

public int Day

{

get { return day; }

set { day = value; }

}

public int Year

{

get { return year; }

set { year = value; }

}

// Method

public void DisplayDate()

{

Console.WriteLine($"{month:D2}/{day:D2}/{year:D4}");

}

}

}

**Code (Book)**

using System;

using System.Collections.Generic;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

namespace Final\_Exam

{

public class Book : ReadingMaterial

{

// Data Members

public string author;

public Date dateOfPublication { get; set; }

public string Author

{

get { return author; }

set { author = value; }

}

public Book(string au, Date dop, string t, double p) : base(t, p)

{

author = au;

dateOfPublication = dop;

}

// Method

public void Donate()

{

Console.WriteLine($"You have donated {tittle} by {author}");

}

public void ReadBook()

{

Console.WriteLine($"You are reading {tittle} by {author}, published on {dateOfPublication}. This item was bought for {price} ");

}

}

}

**Code (Magazine)**

using System;

using System.Collections.Generic;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

namespace Final\_Exam

{

public class Magazine : ReadingMaterial

{

// Data Members

public string editorInChief { get; set; }

// Overload C.

public Magazine (string eic, string t, double p) : base(t, p)

{

editorInChief = eic;

}

// Methods

public void Subscribe()

{

Console.WriteLine($"You are now subscribed to {Tittle}, you will be received updates regarding a sequel if any!");

}

public void Discount(double discount)

{

price -= discount;

}

}

}

**Code (ReadingMaterial)**

using System;

using System.Collections.Generic;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

namespace Final\_Exam

{

public class ReadingMaterial

{

// Data Members

public string tittle;

public double price;

// Default C.

public ReadingMaterial()

{

tittle = "";

price = 0.00;

}

// Overload C.

public ReadingMaterial(string t, double p)

{

tittle = t;

price = p;

}

// Get n Set

public string Tittle

{

get { return tittle; }

set { tittle = value; }

}

public double Price

{

get { return price; }

set { price = value; }

}

// Method

public void Read()

{

Console.WriteLine("You are reading " + tittle);

}

}

}

**UML**

|  |
| --- |
| ReadingMaterial |
|  |
| +tittle: string |
| +price: double |
|  |
| +ReadingMaterial() |
| +ReadingMaterial(string, double) |
| +GetTittle(): string |
| +SetTittle(string) |
| +GetPrice(): double |
| +SetPrice(double) |
| +Read() |

|  |
| --- |
| Date |
|  |
| +month: int |
| +day: int |
| +year: int |
|  |
| +Date() |
| +Date(int, int, int) |
| +GetMonth(): int |
| +SetMonth(int) |
| +GetDay(): int |
| +SetDay(int) |
| +GetYear(): int |
| +SetYear(int) |
| +DisplayDate() |

|  |
| --- |
| Book |
|  |
| +author: string |
| +dateOfPublication: Date |
|  |
| +GetAuthor(): string |
| +SetAuthor(string) |
| +GetdateOfPublication(): Date |
| +SetdateOfPublication(Date) |
| +Book(string, Date) |
| +ReadBook() |
| +Donate() |

|  |
| --- |
| Magazine |
|  |
| +editorInChief: string |
|  |
| +Magazine(string) |
| +Subscribe() |
| +Discount() |

**Notes**

Note that the UML both classes book and magazine are part/base of ReadingMaterial, while Book is also taking an object from date, but is not part of date. (arrow shows)

I did some extra code to display a bit more, hopefully that’s okay. Some other code was simply display what it was given.