

- Voor een geselecteerd lid in de lijst, de details van dit lid weergeeft met de lijst van uitgeleende boeken en de mogelijkheid om een lid toe te voegen, te wijzigen, en de verwijderen
- De lijst van uitleningen te tonen en de mogelijkheid om een uitlening te verwijderen, te wijzigen of toe te voegen

DEEL 2 : Schriftelijk op laptop (open boek)

Beantwoord de opgaven in C#. Maak een Folder

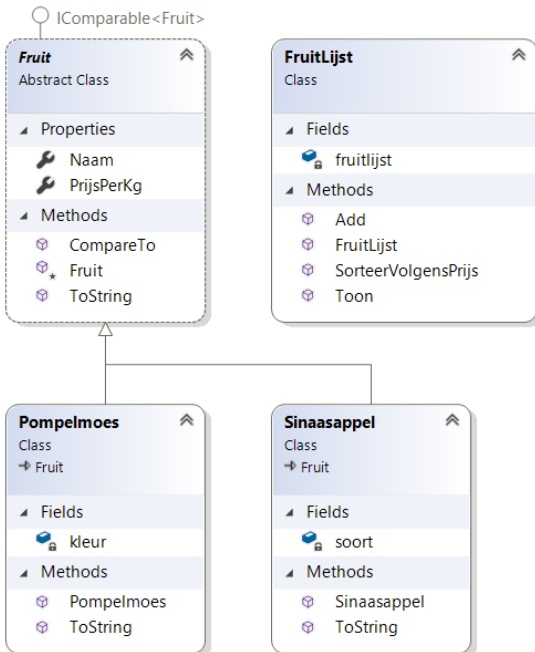
Ex_Programmeren1_FamilieNaam_Voornaam en maak hierin (.Net Core of .Net Framework) (Maak binnen deze folder voor elke Opgave een aparte solution).

Om het resultaat in te dienen, comprimeer (.zip) de volledige solutions folders en uw antwoorden in .doc formaat en plaats deze op uw desktop.

2. OPGAVE OOP – programmatie [/5]

Maak een nieuwe solution en Console applicatie met naam **Opgave_Fruit**

Fruit



De klasse Fruit

De klasse Fruit is een abstracte klasse die tevens de IComparable interface implementeert.

Deze interface is een bestaande interface in .NET, je hoeft hem dus niet zelf te schrijven.

De klasse Fruit bevat een String-Property (Naam) en een double-Property (PrijsPerKg). Naast een constructor die beide Properties invult moet je ook een ToString() methode

voorzien. Werk de methode int CompareTo(Object obj) zo uit dat de PrijsPerKg van de objecten met elkaar vergeleken wordt. Geef 1 terug als de prijs van het this object groter is, -1 als hij kleiner is en 0 als hij gelijk is.

De klasse Citroen

Deze klasse is een uitbreiding van de klasse Fruit. Ze heeft geen extra properties, je hoeft alleen een constructor te voorzien.

De klasse Pompelmoes

De klasse Pompelmoes is een uitbreiding van de klasse Fruit. Ze heeft een extra property Kleur van het type String om de kleur van de pompelmoessoort aan te duiden. Hier moet je naast een constructor ook een ToString methode voorzien.

De klasse SinaasAppel

De klasse SinaasAppel is een uitbreiding van de klasse Fruit. Ze heeft een extra Property van het type bool om aan te duiden of het om een eet- of een persappelsien gaat (true bij eetappelsien). Hier moet je naast een constructor ook een ToString methode voorzien.

De klasse FruitLijst

De klasse FruitLijst bevat slechts één Property, met name een List van Fruitobjecten. Voorzie een constructor om de List fruitlijst te maken.

Zorg voor een Add methode om een Fruit object aan de List toe te voegen.

Voorzie een methode SorteerVolgensPrijs waarbij je gebruik maakt van de methode Sort() van List.

Zorg voor een methode Toon waarmee je alle objecten in fruitlijst afdruckt naar de Console.

Program.cs

Hier mag je niets aan wijzigen. Deze klasse dient alleen om de goede werking van je programma te testen.

```
using System;  
using System.Text;  
using System.Threading;
```

```

namespace Opgave_Fruit_OPL
{
    class Program
    {
        private static Fruit[] fruitsoorten;
        static void Main(string[] args)
        {
            System.Globalization.CultureInfo culture = new
                System.Globalization.CultureInfo("nl-BE");
            Thread.CurrentThread.CurrentCulture = culture;
            Console.OutputEncoding = Encoding.Unicode;
            InitFruitSoorten();
            FruitLijst fruitLijst = new FruitLijst();
            foreach (Fruit fruit in fruitsoorten)
            {
                fruitLijst.Add(fruit);
            }
            fruitLijst.SorteerVolgensPrijs();
            fruitLijst.Toon();
            Console.ReadKey();
        }
        private static void InitFruitSoorten()
        {
            fruitsoorten = new Fruit[] {
                new Citroen("Algarve", 3.0),
                new Sinaasappel("Jaffa", 3.2, true),
                new Sinaasappel("Bloedappelsien", 3.4, true),
                new Pompelmoes("Roze pompelmoes", 3.1, "roze"),
                new Sinaasappel("Persappelsien", 2.5, false),
                new Pompelmoes("Rode pompelmoes", 3.3, "rood"),
                new Pompelmoes("Witte pompelmoes", 2.9, "wit"),
            };
        }
    }
}

```

Hierna volgt de gewenste uitvoer:

```

D:\SYNTRA\CSHARPDEV\CSHARP\2deREEKS\EXAMEN\Opgave_Fruit_OPL\Opgave_Fruit_OPL\bin\Debug\Opgave_Fruit...
Persappelsien : € 2,50 persappelsien
Witte pompelmoes : € 2,90 wit
Algarve : € 3,00
Roze pompelmoes : € 3,10 roze
Jaffa : € 3,20 eetappelsien
Rode pompelmoes : € 3,30 rood
Bloedappelsien : € 3,40 eetappelsien

```

3. OPGAVE WPF – MVVM programmatie [/15]

Open de solution (in .zip) met naam Opgave_WPF_MVVM.

Vul het project aan. Implementeer je ontwerp op papier (klassen en UI) in de Bib- app.

Must have:

- Klassen: implementeer de klassen je hebt gemaakt in Deel 1
- Vul de MockDataService aan zodat er objecten van je klassen worden ingevuld met dummy-gegevens en voeg zo nodig extra definities toe aan IDataService
- WPF UI: een lijst van leden, wanneer op een lid wordt geklikt in de lijst verschijnen de details van het geselecteerde lid met de mogelijkheid om een nieuw lid toe te voegen, de gegevens van een lid te wijzigen en het geselecteerde lid te verwijderen.
- Maak minstens één keer gebruik van elk van de volgende controls: TextBlock, TextBox, DatePicker, ListBox en/of Combobox en datagrid met binding naar de ViewModel properties

Een nice-to-have zou zijn om ook in de lijst van boeken de categorie te zien en op categorie te kunnen filteren.

Maak gebruik van MVVM en RelayCommand en ObservableObject en IDataService. Voorzie in de MockDataService een aantal dummy-leden en uitleningen.