

C# Adv Oefeningen: LINQ

1. Los de volgende vragen op met behulp van een LINQ query.

Gebruik de volgende IEnumerable objecten:

```
List<int> driehoek = new List<int>()
{
    1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1,
    1, 2, 3, 4, 5, 6, 7, 8, 9, 10,
    1, 3, 6, 10, 15, 21, 28, 36, 45,
    1, 4, 10, 20, 35, 56, 84, 120,
    1, 5, 15, 35, 70, 126, 210,
    1, 6, 21, 56, 126, 252,
    1, 7, 28, 84, 210,
    1, 8, 36, 120,
    1, 9, 45,
    1, 10,
    1
};

string[] tekst = new string[]
{
    "langspeelfilm", "vluchtmisdrijf", "dovemansgesprek", "containerpark",
    "confituur", "inox", "valavond", "enerverend", "lopen", "bank", "vieruurtje",
    "waterzooi", "maaltijdcheque", "arrondissement", "eindtermen", "nieuwjaarsbrief"
};
```

- Druk al de woorden af in de array, tekst, die langer zijn dan tien karakters.
- Bereken het gemiddelde voor de getallen in de list driehoek die groter zijn dan 100.
- Druk al de unieke getallen af uit de List, driehoek.
- Druk alle unieke getallen af uit de List, driehoek, die deelbaar zijn door drie of vijf. De getallen worden in een oplopende volgorde afgedrukt.
- Druk de woorden uit de array, tekst, af in een aflopende volgorde op basis van lengte. Verder zijn woorden met dezelfde lengte gesorteerd op een oplopende alfabetische volgorde.
- Maak een groep op basis van de startletter van ieder woord voor al de woorden in de array, tekst.
- Vind alle paren van getallen en woorden waar de lengte van het woord uit de array, tekst, gelijk is aan een getal uit de List, driehoek. Druk het alle gevonden paren af.
- Maak een uitbreiding op de vorige oefening die er voor zorgt dat elk woord slechts één maal voorkomt in de resulterende paren.

- i. Druk alle woorden af uit de array, tekst, die minstens drie klinkers bevatten.
2. Een bedrijf heeft magazijn over heel België. Gebruik LINQ en model classes om hun queries uit te voeren. Maak twee model classes met de volgende properties:
 - Warehouse:
 - BuildingName
 - WarehouseID
 - City
 - PostCode
 - Street
 - HouseNumber
 - StorageCapacity (in m²)
 - EmployeeSatisfactionRating (een lijst van scores van één tot vijf)
 - Employee:
 - FirstName
 - LastName
 - ID
 - WarehouseID

Gebruik de volgende code om data in te laden in je programma en los vervolgens de gevraagde queries op:

```
List<Warehouse> warehouses = new List<Warehouse>
{
    new Warehouse("Brug4", 0, "Arendonk", 2370, "Holstraat", 14, 3000, new List<int>{ 4, 3, 1, 5 }),
    new Warehouse("Brug1", 1, "Arendonk", 2370, "Holstraat", 3, 8000, new List<int>{ 1, 4, 3, 5, 2,
3, 3, 4, 4}),
    new Warehouse("Poort1", 2, "Gent", 9000, "Stropkaai", 12, 7000, new List<int>{ 5, 4, 3, 4, 4}),
    new Warehouse("Rijstellers", 3, "Hasselt", 3500, "Industrielaan", 1, 2500, new List<int>{ 5, 4,
3, 5, 2, 3, 4, 4}),
    new Warehouse("Automa klein", 4, "Berchem", 2600, "Nieuwevaart", 77, 10000, new List<int>{ 4 }),
    new Warehouse("Schuifla", 5, "Hasselt", 3500, "Industrielaan", 15, 1500, new List<int>{ 3, 5, 2
}),
    new Warehouse("Automa groot", 6, "Berchem", 2600, "Nieuwevaart", 76, 30000, new List<int>{ 5 }),
    new Warehouse("Brug2", 7, "Arendonk", 2370, "Molenweg", 7, 3000, new List<int>{ 4, 3, 5, 2 }),
    new Warehouse("Veerhal", 8, "Melle", 9090, "Merelstraat", 48, 500, new List<int>{ 5, 5 }),
    new Warehouse("Poort2", 9, "Gent", 9000, "Burgstraat", 113, 6600, new List<int>{ 1, 2, 1, 1, 2,
3}),
    new Warehouse("D1", 10, "Knokke", 8300, "Vaart", 2, 2200, new List<int>{ 5, 4, 1 }),
    new Warehouse("Brug3", 11, "Arendonk", 2370, "Molenweg", 8, 8000, new List<int>{ 5, 2, 3, 5, 5
}),
    new Warehouse("D2", 12, "Knokke", 8300, "Vaart", 4, 2200, new List<int>{ 2, 3, 4 }),
    new Warehouse("D3", 13, "Knokke", 8300, "Vaart", 6, 2200, new List<int>{ 3, 4, 3 })
};

List<Employee> employees = new List<Employee>
{
    new Employee("Jos", "Jansen", 0, 1),
```

```

new Employee("Ted", "Béria ult", 1, 0),
new Employee("Tony", "Hawk", 2, 3),
new Employee("Peggy", "Corona", 3, 12),
new Employee("Edna", "Goosen", 4, 0),
new Employee("Mac", "Kowalski", 5, 11),
new Employee("Alejandro", "Mendoza", 6, 8),
new Employee("Aysha", "Lyon", 7, 7),
new Employee("Tyson", "Dyer", 8, 4),
new Employee("Nanou", "Hahn", 9, 6),
new Employee("Kevin", "Hahn", 10, 5),
new Employee("Kris", "Jacobsen", 11, 1),
new Employee("Boros", "Orzsebet", 12, 2),
new Employee("Buday", "Gedeon", 13, 2),
new Employee("Szölôsi", "Taksony", 14, 1),
new Employee("Kocsis", "Gyula", 15, 8),
new Employee("Asif", "Atiyeh", 16, 7),
new Employee("Ruwayd", "Akram", 17, 13),
new Employee("Makary", "Sobczak", 18, 12),
new Employee("Pawel", "Symanski", 19, 1),
new Employee("Settimio", "Calabresi", 20, 10),
new Employee("Ivo", "Bellucci", 21, 7),
new Employee("Matthieu", "Camus", 22, 9),
new Employee("Jacques", "Huard", 23, 8),
new Employee("Melville", "Béria ult", 24, 4),
new Employee("René", "Michaud", 25, 9)
};

```

3. Los de volgende verzoeken op met behulp van LINQ

- Wat zijn de namen van de magazijnen in Berchem?
- Wat zijn de namen en steden van al de magazijnen geordend (van hoog naar laag) op meeste ratings van hun werknemers
- Wat zijn de volledige namen en id van al de werknemers alfabetisch geordend op hun achternaam? (a tot z)
- Rangschik al de magazijnen van hoog naar laag op basis van gemiddelde rating.
- Wat zijn de magazijnen met postcode onder 4000 gegroepeerd per stad ?
- Wat zijn de volledige namen van de werknemers die werken voor een magazijn met een opslagcapaciteit groter dan 2000? Vermeld ook de naam en capaciteit van het magazijn.
- Wat zijn de id's en voornamen van werknemers die een achternaam delen met een andere werknemer van het magazijn?
- Wat zijn de voornamen van de werknemers die werken voor een magazijn met een opslagcapaciteit die groter is dan 5000. Vermeld ook de volledige locatie (stad, postcode, straat, huisnummer) van het magazijn.
- Groep de werknemers per straat van het magazijn waarvoor ze werken.

