

Övning Modul 6B

ArrayList

Arbetsuppgift 1: Skapa konsolprogram

Steg 1: Klicka på New – Project, välj därefter Console Application.

Steg 2: Skriv in programkod enligt nedan:

```
using System;
using System.Collections;
using System.Linq;
using System.Text;
using System.Threading.Tasks;

namespace ArrayList1
{
    public struct Coffee
    {
        public Coffee(int strength, string bean, string countryOfOrigin)
        {
            this.Strength = strength;
            this.Bean = bean;
            this.CountryOfOrigin = countryOfOrigin;
        }
        public int Strength;
        public string Bean;
        public string CountryOfOrigin;
    }

    class Program
    {
        static void Main(string[] args)
        {
            // Skapa nya ArrayList collection.
            ArrayList beverages = new ArrayList();
            // Skapa poster för att läggas till collection.
            Coffee coffee1 = new Coffee(4, "Arabica", "Columbia");
            Coffee coffee2 = new Coffee(3, "Arabica", "Vietnam");
            Coffee coffee3 = new Coffee(4, "Robusta", "Indonesia");
            // Lägg till collection.
            // Poster konverteras till Object type när du lägger till dessa.
            beverages.Add(coffee1);
            beverages.Add(coffee2);
            beverages.Add(coffee3);
            // Hämtar från collection.
            // Måste konverteras (cast) tillbaka till original typ.
            Coffee firstCoffee = (Coffee)beverages[0];
            Coffee secondCoffee = (Coffee)beverages[1];

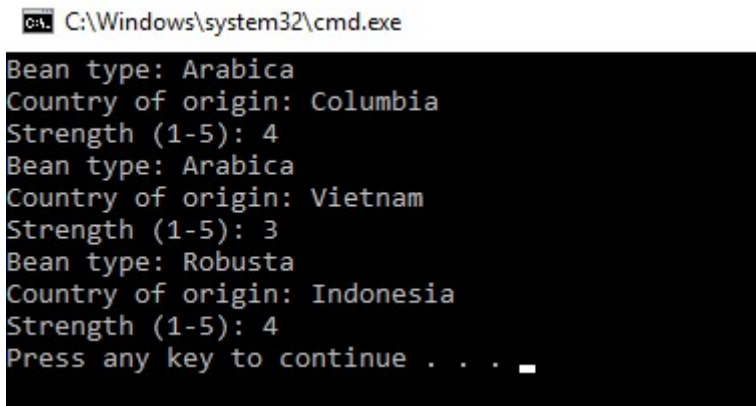
            foreach (Coffee coffee in beverages)
            {
                Console.WriteLine("Bean type: {0}", coffee.Bean);
                Console.WriteLine("Country of origin: {0}", coffee.CountryOfOrigin);
                Console.WriteLine("Strength (1-5): {0}", coffee.Strength);
            }
        }
    }
}
```

```

    }
  }
}

```

Steg 3: Klicka på Debug – Start Without Debugging.



```

C:\Windows\system32\cmd.exe
Bean type: Arabica
Country of origin: Columbia
Strength (1-5): 4
Bean type: Arabica
Country of origin: Vietnam
Strength (1-5): 3
Bean type: Robusta
Country of origin: Indonesia
Strength (1-5): 4
Press any key to continue . . .

```

HashTable

Arbetsuppgift 1: Skapa konsolprogram

Steg 1: Klicka på New – Project, välj därefter Console Application.

Steg 2: Skriv in programkod enligt nedan:

```

using System;
using System.Collections;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using System.Threading.Tasks;

namespace CoffeeHashTable
{
    class Program
    {
        static void Main(string[] args)
        {
            // Skapa ny Hashtable collection.
            Hashtable ingredients = new Hashtable();
            // Lägg till par bestående av nyckel/värde till samling.
            ingredients.Add("Café au Lait", "Coffee, Milk");
            ingredients.Add("Café Mocha", "Coffee, Milk, Chocolate");
            ingredients.Add("Cappuccino", "Coffee, Milk, Foam");
            ingredients.Add("Irish Coffee", "Coffee, Whiskey, Cream, Sugar");
            ingredients.Add("Macchiato", "Coffee, Milk, Foam");
            // Kontrollera om nyckel finns.
            if (ingredients.ContainsKey("Irish Coffee"))
            {

```

```

        // Hämta värde som är knytet till nyckel.
        Console.WriteLine("The ingredients of a Irish Coffee are: {0}",
ingredients["Irish Coffee"]);
    }
}
}
}
}

```

Steg 3: Klicka på Debug – Start Without Debugging.

C:\Windows\system32\cmd.exe

```

The ingredients of a Irish Coffee are: Coffee, Whiskey, Cream, Sugar
Press any key to continue . . .

```

LINQ

Arbetsuppgift 1: Skapa konsolprogram

Steg 1: Klicka på New – Project, välj därefter Console Application.

Steg 2: Skriv in programkod enligt nedan:

```

using System;
using System.Collections;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using System.Threading.Tasks;

namespace CoffeeLINQ
{
    class Program
    {
        static void Main(string[] args)
        {
            // Skapa Hashtable och lägg till lite dricka med priser.
            // M eller m anger att det skall vara decimaler
            Hashtable prices = new Hashtable();
            prices.Add("Café au Lait", 1.99M);
            prices.Add("Caffe Americano", 1.89M);
            prices.Add("Café Mocha", 2.99M);
            prices.Add("Cappuccino", 2.49M);
            prices.Add("Espresso", 1.49M);
            prices.Add("Espresso Romano", 1.59M);
            prices.Add("English Tea", 1.69M);
            prices.Add("Juice", 2.89M);
            // Välj dricka som kostar mindre än $2.00 och sortera dessa efter kostnad.
            var bargains =
            from string drink in prices.Keys
            where (Decimal)prices[drink] < 2.00M
            orderby prices[drink] ascending
            select drink;
            // Visa resultatet.

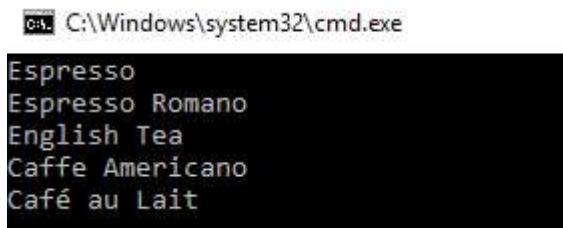
```

```

        foreach (string bargain in bargains)
        {
            Console.WriteLine(bargain);
        }
        Console.ReadLine();
    }
}

```

Steg 3: Klicka på Debug – Start Without Debugging.



```

C:\Windows\system32\cmd.exe
Espresso
Espresso Romano
English Tea
Caffè Americano
Café au Lait

```

Arbetsuppgift 2: Skapa konsolprogram

Steg 1: Klicka på New – Project, välj därefter Console Application.

Steg 2: Skriv in programkod enligt nedan:

```

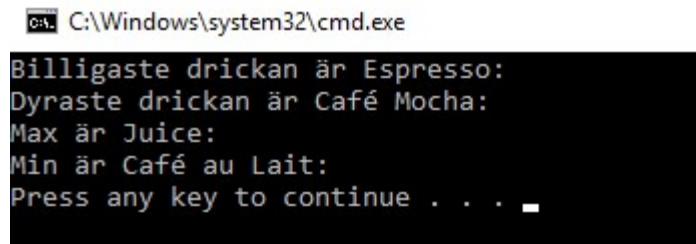
using System;
using System.Collections;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using System.Threading.Tasks;

namespace LINQExtra
{
    class Program
    {
        static void Main(string[] args)
        {
            // Skapa Hashtable och lägg till lite dricka med priser.
            Hashtable prices = new Hashtable();
            prices.Add("Café au Lait", 1.99M);
            prices.Add("Caffè Americano", 1.89M);
            prices.Add("Café Mocha", 2.99M);
            prices.Add("Cappuccino", 2.49M);
            prices.Add("Espresso", 1.49M);
            prices.Add("Espresso Romano", 1.59M);
            prices.Add("English Tea", 1.69M);
            prices.Add("Juice", 2.89M);
            // Fråga Hashtable och sortera dricka efter kostnad.
            var drinks =
            from string drink in prices.Keys
            orderby prices[drink] ascending // sorteringsordning
            select drink;
            Console.WriteLine("Billigaste drickan är {0}: ", drinks.FirstOrDefault());
            // Output: "Billigaste drickan är Espresso"
            Console.WriteLine("Dyraste drickan är {0}: ", drinks.Last());
            // Output: "Dyrast drickan är Café Mocha"
            Console.WriteLine("Max är {0}: ", drinks.Max());
        }
    }
}

```

```
        // Output: "Max är Juice"  
        // "Juice" har det högsta värdet i samlingen, när den sorteras  
alfabetiskt.  
        Console.WriteLine("Min är {0}: ", drinks.Min());  
        // Output: "Min är Café au Lait"  
        // "Café au Lait" har det lägsta värdet i samlingen, när denna sorteras  
alfabetiskt.  
    }  
}
```

Steg 3: Klicka på Debug – Start Without Debugging.



```
C:\Windows\system32\cmd.exe  
Billigaste drickan är Espresso:  
Dyraste drickan är Café Mocha:  
Max är Juice:  
Min är Café au Lait:  
Press any key to continue . . . _
```