Övningar för Modul 4a

Övning Selections

Övning 1

Ladda ner övningsfiler.

Steg 1: Öppna Internet Explorer, skriv in följande URL: https://easec.se/c/Selection.zip, välj att spara filen i katalogen c:\easec.

(OBS! Finns inte katalogen får du skapa den!)

Steg 2: Förflytta dig till katalogen C:\easec, högerklicka på Selection.zip och välj alternativet Extract All. Välj att packa upp till C:\easec\Modul 1. Klicka på Extract.

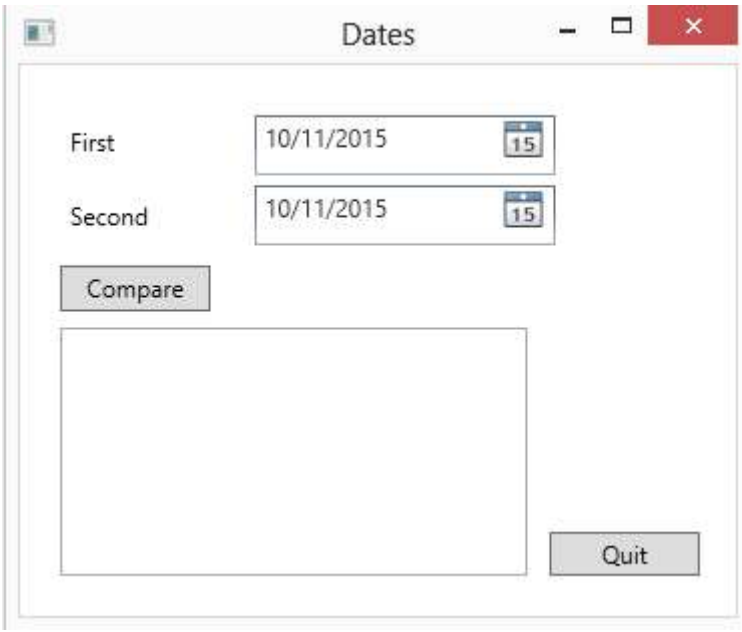
Övning 2

Steg 1: Starta Visual Studio.

Steg 2: Öppna lösningen: \easec\Modul 1\Selection\Selection.sln.

Steg 3: Klicka på Start Debugging, via DEBUG meny.

Följande visas:



Du har två stycken fält, som bägge visar nuvarande datum. Dessa kallas First och Second. Datum kan justeras genom att klicka på ikon för respektive datumfält.

Steg 4: Klicka på Compare, utan att ändra datum.

I nedre delen kommer följande att visas:

first == second : False

first != second : True

first < second : False

first <= second : False

first > second : True

first >= second : True

Boolean uttrycket first == second borde vara True eftersom både first och second är satt till nuvarande datum.

Bara mindre än, större än och lika med ser ut att fungera korrekt.

Steg 5: Återvänd till Visual Studio, från meny DEBUG klicka på Stop Debugging.

Steg 6: I Solution Explorer, dubbelklicka på MainWindow.xaml.cs, så koden visas i fönstret Code and Text Editor.

Steg 7: Lokalisera metoden compareClick, den ser ut så här:

```
private void compareClick(object sender,
RoutedEventArgs e)

{

int diff = dateCompare(first.SelectedDate.Value,
second.SelectedDate.Value);

info.Text = "";

show("first == second", diff == 0);
```

```
show("first != second", diff != 0);

show("first < second", diff < 0);

show("first <= second", diff <= 0);

show("first > second", diff > 0);

show("first >= second", diff >= 0);

}
```

Metoden körs varje gång användare klickar på knappen Compare.

Variablarna first och second innehåller värdena för DateTime och får värden när värden visas i de två fälten i formuläret, i en annan del av programmet.

DateTime är ett exempel på data type som innehåller underelement såsom år, månad eller dag, som gör det möjligt att få åtkomst till dessa individuellt.

Metoden compareClick skickar vidare värdet för DateTime till metoden dateCompare. Syftet med denna metod är att jämföra datum och skicka tillbaks värdet 0 (int) om värdena är samma, -1 om första datumet är mindre än det andra och +1 om första datumet är större än det andra.

Datum beräknas som större än det andra om det kommer kronologiskt efter. Du kommer att undersöka metoden i nästa steg.

Metoden show visar resultat av jämförelsen i boxen infotext, denna finns i den nedre delen av formuläret.

Steg 8: Lokalisera metoden dateCompare, den ser ut på detta sätt:

```
private int dateCompare(DateTime leftHandSide,
DateTime rightHandSide)

{

// TO DO

return 42;

}
```

Metoden returnerar nu alltid samma värde, oavsett hur den kallas. Egentligen skall den returnera 0, -1 eller +1. Detta förklarar varför programmet inte fungerar som förväntat!

Du kommer att behöva implementera logik i denna metod så metoden kan jämföra två datum på ett korrekt sätt.

Steg 9: Plocka bort kommentaren // TO DO och return 42;, från metoden.

Steg 10: Lägg till följande rader som visas med fet text:

```
private int dateCompare(DateTime leftHandSide,
DateTime rightHandSide)
{
int result;
if (leftHandSide.Year < rightHandSide.Year)
{
result = -1;
}
else if (leftHandSide.Year > rightHandSide.Year)
{
result = 1;
}
}
```

Om uttrycket leftHandSide.Year < rightHandSide.Year är sant (true), så måste datumet i leftHandSide vara tidigare än rightHandSide, kommer programmet att sätta variabel för result till -1, annars kommer uttrycket leftHandSide.Year > rightHandSide vara sant och variabel för result kommer att sättas till 1.

Om uttrycket leftHandSide.Year < rightHandSide.Year är falskt och utrycket leftHandSide.Year > rightHandSide.Year är falskt, måste egenskap för Year vara samma, i så fall måste programmet undersöka månad för varje datum.

Steg 11: Lägg till följande uttryck, som visas med fet text, i metoden dateCompare, efter kod som du skrev in tidigare:

```
private int dateCompare(DateTime leftHandSide,
DateTime rightHandSide)

{

...

else if (leftHandSide.Month < rightHandSide.Month)

{

result = -1;

}

else if (leftHandSide.Month > rightHandSide.Month)

{

result = 1;

}

}
```

Koden som du la till, följer liknade logik för att jämföra månader som du implementerade för att jämföra år.

Om uttrycket leftHandSide.Month < rightHandSide.Month är falskt (false) och uttrycket leftHandSide.Month > rightHandSide.Month också är falskt, måste värdet vara detsamma i fältet datum, så programmet måste tillslut även jämföra fältet för dag i varje datum.

Steg 12: Lägg till följande uttryck, som visas med fet text, i metoden dateCompare, efter kod som du skrev in tidigare:

```
private int dateCompare(DateTime leftHandSide,
DateTime rightHandSide)

{

...
```

```
else if (leftHandSide.Day < rightHandSide.Day)
{
result = -1;
}
else if (leftHandSide.Day > rightHandSide.Day)
{
result = 1;
}
else
{
result = 0;
}
return result;
}
```

Du känner igen mönstret sedan tidigare. Om leftHandSide.Day < rightHandSide.Day och leftHandSide.Day > rightHandSide.Day bägge är falska, så måste värdet för egenskap för dag vara samma i både variablarna.

Värdet för månad och år, måste också vara samma om logik för programmet har kommit så här långt, programmet kommer att sätta värdet i result till 0.

Det sista uttrycket returnerar värdet som är lagrat i variabel för result.

Steg 13: Klicka på Start Debugging, från DEBUG meny. Fältet för first och second är satt till nuvarande datum.

Steg 14: Klicka på Compare.

Följande text kommer att dyka upp i den nedre delen av formuläret:

first == second : True

first != second : False

first < second: False

first <= second: True

first > second: False

first >= second: True

Detta är det korrekta resultatet för identiska datum.

Steg 15: Ändra datumet i fältet för second, till några dagar längre fram.

Steg 16: Klicka på Compare.

Följande text kommer att dyka upp i den nedre delen av formuläret:

first == second: False

first != second: True

first < second: True

first <= second: True

first > second: False

first >= second: False

Detta är det korrekta resultat när första datumet är tidigare än den andra.

Steg 17: Testa några andra kombinationer för de olika fältet, verifiera att resultaten blir som du har förväntat dig.

Steg 18: Återvänd till Visual Studio och klicka på alternativet Stop Debugging.

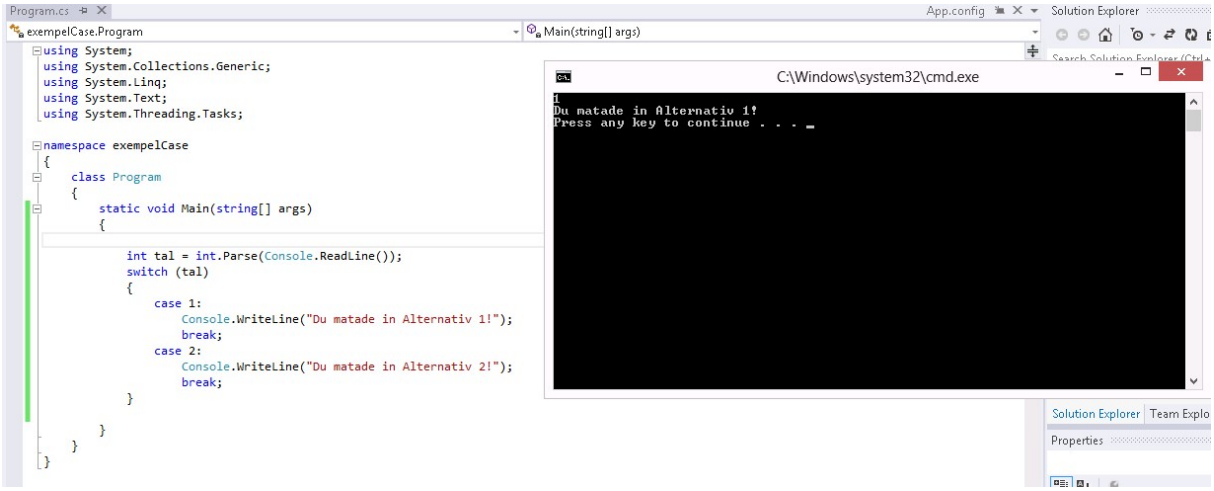
Du är klar!

Övning switch

Switch-satsen påminner om if-satsen, men är speciellt konstruerad för att hantera uppräkningar av olika utfall. I satsen definieras en eller flera case-satser som sedan avslutas med nyckelordet break.

Exempel:

```
int tal = int.Parse(Console.ReadLine());
switch(tal)
{
   case 1:
        Console.WriteLine(”Alternativ 1 matades in!”);
        break;
   case 2:
        Console.WriteLine(”Alternativ 2 matades in!”);
        break;
}
```



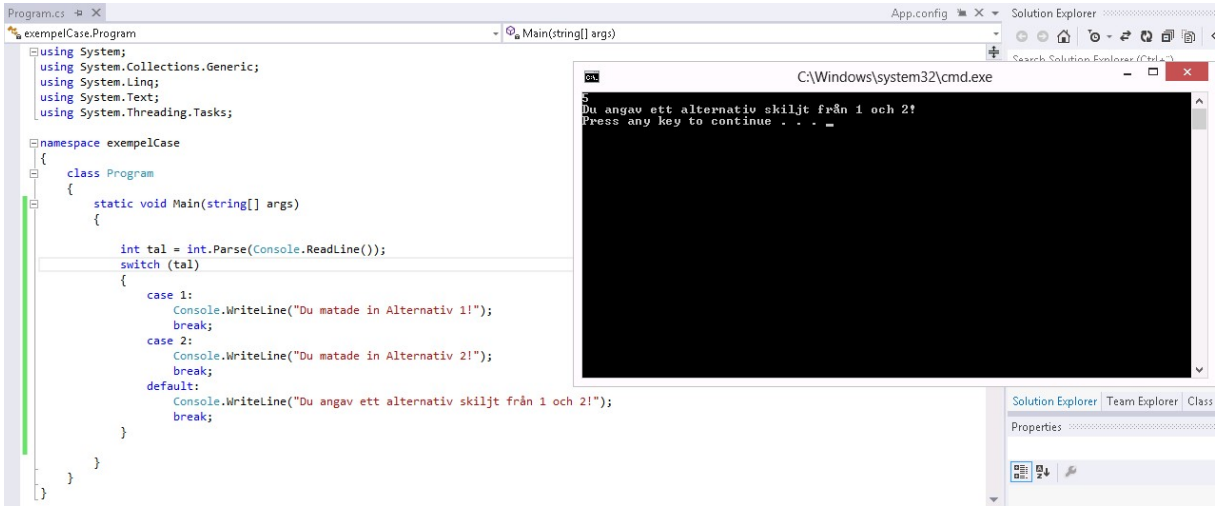
Om 1 eller 2 matas in, skrivs texten ut. Om annat tal skrivs in, kommer däremot ingen text att skrivas ut.

Varje case följs av ett värde som direkt jämförs med det uttryck som switch-satsen hanterar.

I exemplet används variablen tal, efter värdet följs ett ; och sedan koden som körs när fallet (case) inträffar. Flera rader av kod kan användas, men fallet måste avslutas med ett break och ett semikolon.

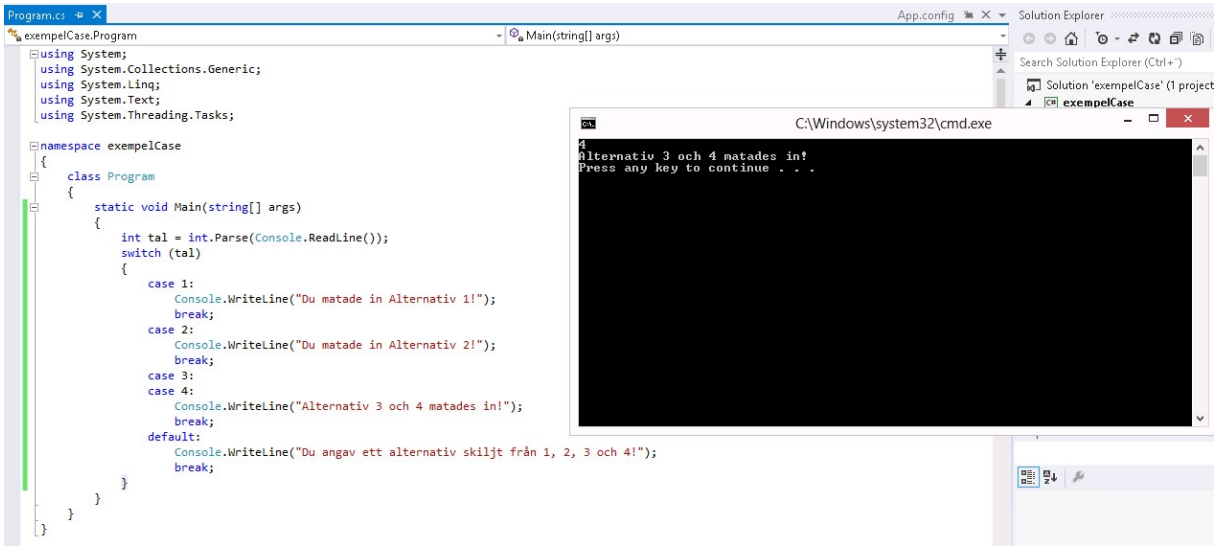
Om du kompletterar ditt program enligt fet text, kommer allt som är skiljt från 1 eller 2 att tas omhand.

```
int tal = int.Parse(Console.ReadLine());
switch(tal)
{
    case 1:
        Console.WriteLine(”Alternativ 1 matades in!”);
        break;
    case 2:
        Console.WriteLine(”Alternativ 2 matades in!”);
        break;
    default:
        Console.WriteLine(”Du angav ett alternativ skiljt från 1 och 2!”);
        break;
}
```



Ibland kan det vara så att två eller flera fall resulterar i samma kod. Detta löses att i en swich-sats skriva på följande sätt:

```
int tal = int.Parse(Console.ReadLine());
switch(tal)
{
   case 1:
        Console.WriteLine(”Alternativ 1 matades in!”);
        break;
   case 2:
        Console.WriteLine(”Alternativ 2 matades in!”);
        break;
   case 3:
   case 4:
        Console.WriteLine(”Alternativ 3 och 4 matades
in!”);
        break;
   default:
        Console.WriteLine(”Du angav ett alternativ
skiljt från 1, 2, 3 och 4!”);
        break;
}
```



Övning 1

Steg 1: På din virtuella maskin, starta Visual Studio.

Steg 2: Klicka på File-New-Project. I den vänstra trädstrukturen klicka på Visual C#. Klicka därefter på Console Application i det mittersta fönstret.

Steg 3: Skriv in SwitchÖvning1 i rutan till höger om File Name, klicka därefter på OK.

Steg 4: Skriv in följande kod som är markerad med fet text.

```
namespace SwitchÖvning1
{
        Class Program
        {
            static void Main(string[] args)
            {
                Console.Write(”Ange ett betyg (A-F):
”);
                string betyg = Console.ReadLine();
                switch (betyg)
                {
```

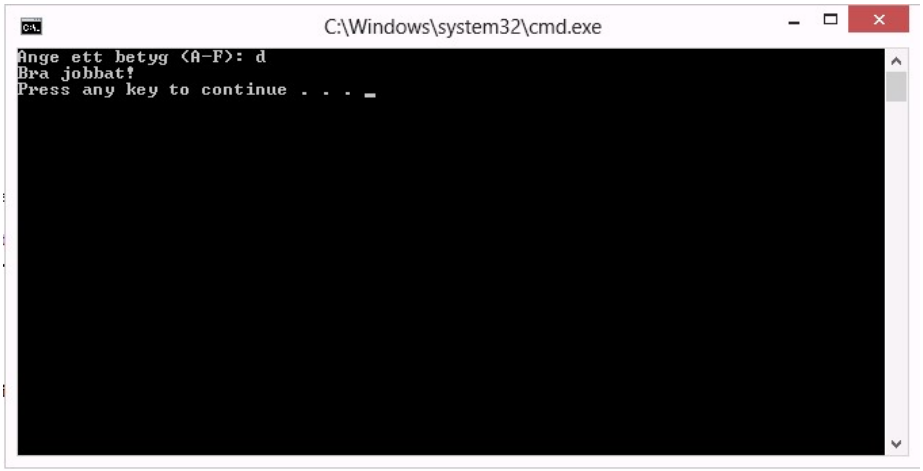
```
                    case ”A”:
                    case ”a”:
                        Console.WriteLine(”Perfekt, du
är grym!”);
                        break;
                    case ”B”:
                    case ”b”:
                        Console.WriteLine(”Bra
jobbat!”);
                        break;
                    case ”C”:
                    case ”c”:
                        Console.WriteLine(”Bra jobbat,
lite mer kan du!”);
                        break;
                    case ”D”:
                    case ”d”:
                        Console.WriteLine(”Bra
jobbat!”);
                        break;
                    case ”E”:
                    case ”e”:
                        Console.WriteLine(”Du är
godkänd!”);
                        break;
                    case ”F”:
                    case ”f”:
```

```
                    Console.WriteLine(”Du har lite
att arbeta med!”);
                    break;
                default:
                    Console.WriteLine(”Du har
skrivit in fel betyg!”);
                    break;
                }
            }
        }
}
```

```
Program.cs*
SwitchÖvning1.Program
Main(string[] args)
        {
            Console.Write("Ange ett betyg (A-F): ");
            string betyg = Console.ReadLine();

            switch (betyg)
            {
                case "A":
                case "a":
                    Console.WriteLine("Perfekt, du är grym!");
                    break;
                case "B":
                case "b":
                    Console.WriteLine("Bra jobbat!");
                    break;
                case "C":
                case "c":
                    Console.WriteLine("Bra jobbat, lite mer kan du!");
                    break;
                case "D":
                case "d":
                    Console.WriteLine("Bra jobbat!");
                    break;
                case "E":
                case "e":
                    Console.WriteLine("Du är godkänd!");
                    break;
                case "F":
                case "f":
                    Console.WriteLine("Du har lite att arbeta med!");
                    break;
                default:
                    Console.WriteLine("Du har skrivit in fel betyg!");
                    break;
```

Steg 5: Klicka på DEBUG-Start Without Debugging.



Steg 6: Verifiera att ditt program fungerar, klicka på någon tangent för att avsluta ditt program.

Gör övningarna nedan, döp de olika övningarna till SwitchÖvningX.

Övning 2

Skriv ett program där du får ange ett alternativ i form av ett heltal. Beroende på alternativ så skall olika saker skrivas ut enligt nedan:

1: ”Ett”

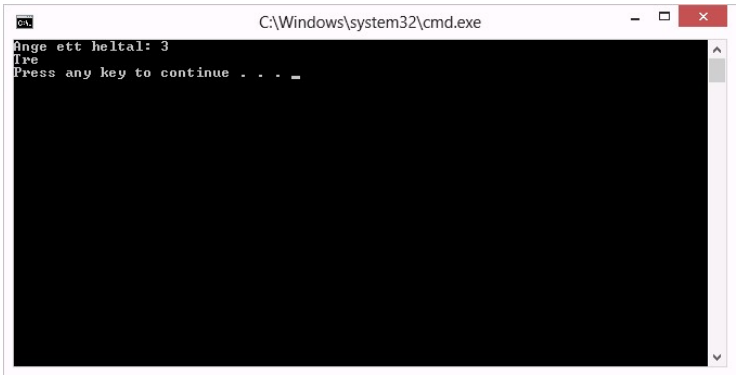
2: ”Två”

3: ”Tre”

4: ”Fyra”

För övriga alternativ, skall programmet skriva ut ”Ogiltigt alternativ!”.

Se bild nedan:



Övning 3

Skriv ett litet textäventyr. Du matar in en bokstav (som symboliserar väderstreck), ditt program reagerar enligt nedan:

N (eller n): ”Du färdas norrut”

S (eller s): ”Du färdas söderut”

Ö (eller ö): ”Din vandring tar dig österut”

V (eller v): ”Din vandring tar dig västerut”

Övriga alternativ: ”Jag förstår inte!”

Se bild nedan: