

Objektorienterad programmering

Föreläsning 3

© Copyright
Mahmud Al Hakim
mahmud@webacademy.se
www.webacademy.se

Agenda (obs! halvdag)

- Do-satsen
- Break-satsen
- Continue-satsen
- Goto-satsen
- Switch-satsen
- Villkorsoperatörn

Do-satsen

- Do-satsen liknar while-satsen men skillnaden är att testuttrycket beräknas och testas efter varje varv istället för före.
- Detta innebär satser inne i do-satsen alltid utförs minst en gång.

```
do{  
    satser ...  
} while (uttryck)
```

Do-satsen – Exempel

```
do  
{  
    Console.Write("Skriv lite text: ");  
  
    Console.WriteLine("Antal tecken: " +  
        Console.ReadLine().Length);  
  
    Console.Write("Vill du fortsätta (skriv Ja eller J)? ");  
} while (Console.ReadLine().ToLower().StartsWith("j"));
```

Break-satsen

- Break-satsen används för att hoppa ut ur en repetitionssats (while, for eller do) eller en switch-sats.
- Skriver man bara break så hoppar man ur den närmast omgivande satsen.
- Hoppet sker alltid till första sats som ligger direkt efter den sats man hoppar ur.

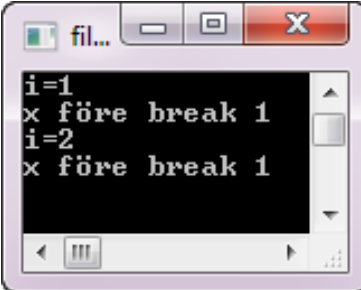
Break-satsen – Exempel

```
for (int i = 1; i <= 2; i++)
{
    Console.WriteLine("i="+i);

    for (int x = 1; x <= 2; x++)
    {
        Console.WriteLine("x före break " + x);

        if (x == 1)
            break;

        Console.WriteLine("x efter break " + x);
    }
}
```



```
i=1
x före break 1
i=2
x före break 1
```

Continue-satsen

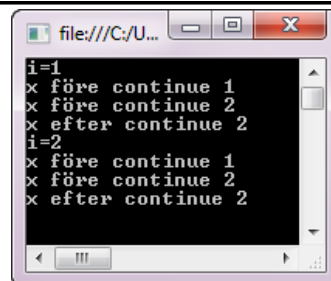
- En continue-sats får bara finnas inne i en repetitionssats.
- Continue-satsen avbryter , liksom break-satsen exekveringen på det aktuella stället och gör ett hopp!
- Det aktuella varvet i repetitionssatsen avbryts och exekveringen kommer direkt att fortsätta med nästa varv.

Continue-satsen – Exempel

```
for (int i = 1; i <= 2; i++)
{
    Console.WriteLine("i="+i);

    for (int x = 1; x <= 2; x++)
    {
        Console.WriteLine("x före continue " + x);
        if (x == 1)
            continue;

        Console.WriteLine("x efter continue " + x);
    }
}
```



```
file:///C:/U...
i=1
x före continue 1
x före continue 2
x efter continue 2
i=2
x före continue 1
x före continue 2
x efter continue 2
```

Goto-satsen

- Varje sats i C# kan förses med en etikett (label på engelska).
- En etikett är ett slags namn som skrivs framför satsen t.ex.
start: string indata = Console.ReadLine();
- Man kan använda en goto-sats för att hoppa till en viss sats t.ex.
goto start;
- Tips! Goto skapar s.k. "Spaghetti code" så undvik denna sats så gott du kan.

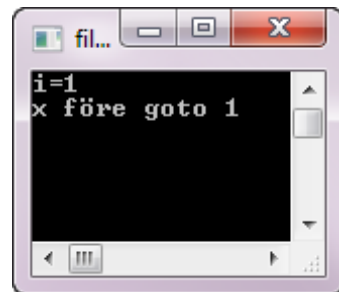
Goto-satsen – Exempel

```
for (int i = 1; i <= 2; i++)
{
    Console.WriteLine("i=" + i);

    for (int x = 1; x <= 2; x++)
    {
        Console.WriteLine("x före goto " + x);

        if (x == 1)
            goto slut;

        Console.WriteLine("x efter goto " + x);
    }
}
slut:
```



Switch-satsen

- En switch-sats är en sats som kan användas istället för if-satsen när man har flervalssituation.
- En switch-sats inleds med det reserverade ordet **switch** och direkt efteråt ska ett **testuttryck** stå inom parentes. Testuttrycket måste vara av heltalstyp, av typen char eller av uppräkningsstyp.
- Varje alternativ i en switch-sats inleds med ordet **case**. Uttrycken som står efter case (case-uttrycken) måste vara konstanta. Det får inte finnas två eller flera case-uttryck med samma värde.
- Det får också eventuellt finnas ett **default**-alternativ.
- En **break**-sats läggs sist i varje alternativ för att hoppa till slutet av switch-satsen.

Switch-satsen – Exempel En enkel kalkylator del I

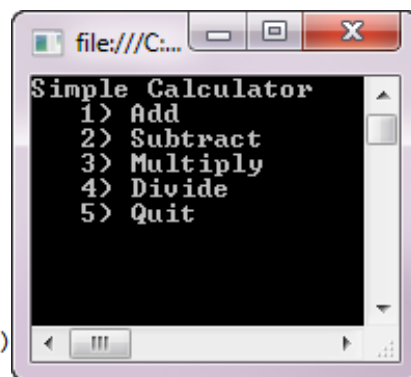
```
int val = 0;
double x = 0;
double y = 0;

Console.WriteLine("Simple Calculator ");
Console.WriteLine(" 1) Add      ");
Console.WriteLine(" 2) Subtract ");
Console.WriteLine(" 3) Multiply ");
Console.WriteLine(" 4) Divide  ");
Console.WriteLine(" 5) Quit   ");

val = Convert.ToInt32(Console.ReadLine());

Console.Write("Enter Number 1: ");
x = double.Parse(Console.ReadLine());

Console.Write("Enter Number 2: ");
y = double.Parse(Console.ReadLine());
```



Switch-satsen – Exempel

En enkel kalkylator del 2

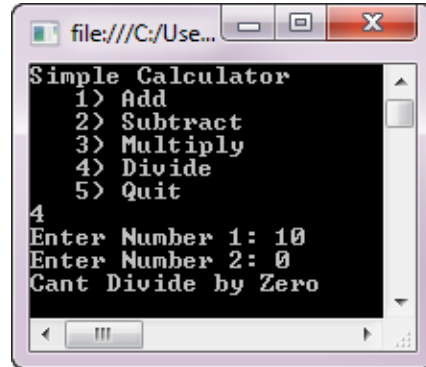
```
switch (val)
{
    case 1:
        Console.WriteLine(x + y);
        break;

    case 2:
        Console.WriteLine(x - y);
        break;

    case 3:
        Console.WriteLine(x * y);
        break;

    case 4:
        if (y == 0)
            Console.WriteLine("Cant Divide by Zero");
        else
            Console.WriteLine(x / y);
        break;

    case 5:
        break;
}
```



Copyright 2015 - Mahmud Al Hakim www.webacademy.se

13

Villkorsoperatoren

Uttryck 1 ? Uttryck 2 : Uttryck 3

Om uttryck 1 är true blir resultatet lika med uttryck 2

Om uttryck 1 är false blir resultatet lika med uttryck 3

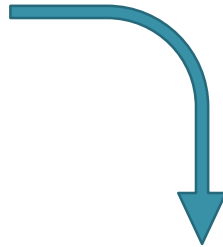
uttryck 2 och uttryck 3 bör ha samma typ.

Copyright 2015 - Mahmud Al Hakim www.webacademy.se

14

Villkorsoperatoren – exempel

```
if (x>y)
    z = x;
else
    z = y;
```



```
z = (x>y) ? X : y;
```

Villkorsoperatoren – Övning

- Skriv ett program som läser in text från kommandotolken.
- Programmet letar efter alla blanksteg och ersätter dessa med semikolon.
- Använd villkorsoperatoren.

A screenshot of a console application window. The title bar shows the file path: file:///C:/oop/ConsoleApplication1/ConsoleApplicat... The window content shows the text: "Mata in lite text!" followed by the input "1 2 3 4 5 A B C" and the output "1;2;3;4;5;A;B;C_".

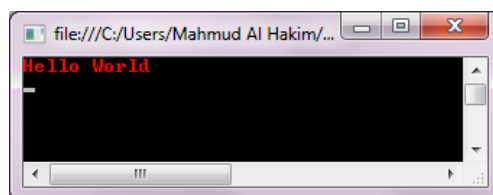
```
file:///C:/oop/ConsoleApplication1/ConsoleApplicat... - □ ×  
Mata in lite text!  
1 2 3 4 5 A B C  
1;2;3;4;5;A;B;C_
```


Lösning

```
Console.WriteLine("Mata in lite text!");  
string s = Console.ReadLine();  
for (int i = 0; i < s.Length; i++)  
    Console.Write( s[i] == ' ' ? ' ' : s[i]);
```

Extra övning I

- Skapa ett C#-program som visar texten "Hello World" i kommandofönstret. Visa texten i valfri förgrundsfärg. Ändra även bakgrundsfärgen.
- Tips: För att ändra textens färg använd metoden `ForegroundColor` som finns i `Console`. Ett antal standardfärger finns i `ConsoleColor` t.ex. `ConsoleColor.Red`
Läs mer på MSDN <http://msdn.microsoft.com>



Extra övning 2

- **Skriv ett program som frågar efter en cirkels radie och beräknar cirkelns area och omkrets!**
- Använd dialogrutor för att läsa indata och presentera resultatet.
- Kontrollera att radien är större än 0 innan du gör beräkningar. Om så inte är fallet skall användaren uppmanas att mata in korrekt värde. Detta förfarande skall upprepas tills ett korrekt värde har matats in.

$$\text{Arean} = \pi r^2 = \frac{\pi d^2}{4}$$

$$\text{Omkretsen} = 2\pi r = \pi d$$