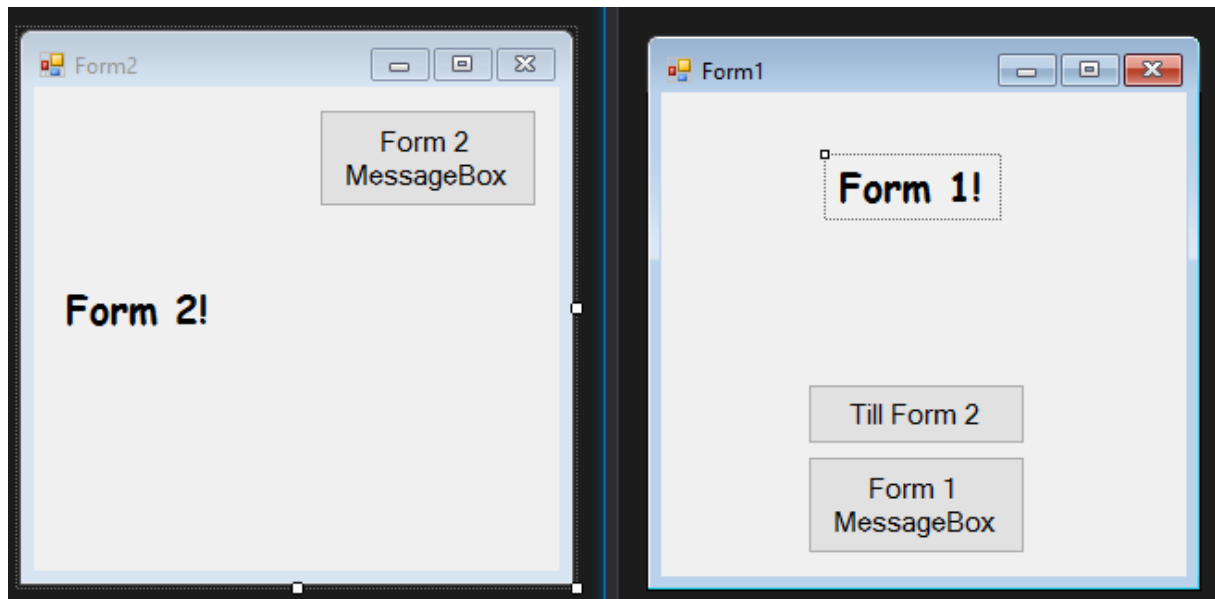


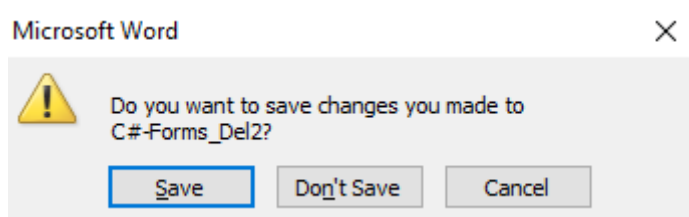
C# - Forms del 2

I denna sektion ska vi gå igenom användandet av MessageBoxes, DialogResult och hur man öppnar nya forms i kod. För dessa exempel har jag skapat två otroligt väldesignade forms, Form1 och Form2. Form 1 består av två knappar, en som ska öppna Form2 och en för att öppna en MessageBox. Form 1 består av en knapp som ska öppna en annan MessageBox.



MessageBox

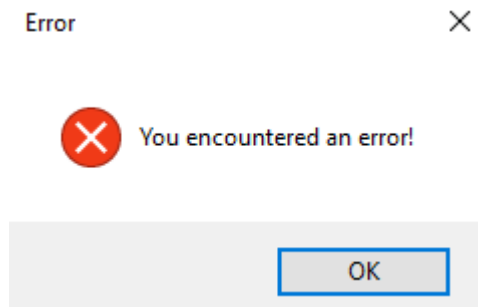
En MessageBox är ett fönster som dyker upp och visar information och ger användaren ett val. Ett exempel på en MessageBox dyker upp när jag försöker trycka ner det här dokumentet då dyker detta upp:



Denna MessageBox fungerar som en varning för användaren när den försöker stänga ner ett osparat dokument. Här får vi tre val, Spara och stänga ner, låta bli att spara och stänga ner, eller avbryt.

För att skapa en MessageBox behöver vi använda oss av objektet "MessageBox" och anropa metoden Show. Därefter matar vi in en sträng för textmeddelande, en sträng för fönstertitel, vilka knappar som ska användas och vilken ikon som ska användas. Koden kan se ut så här:

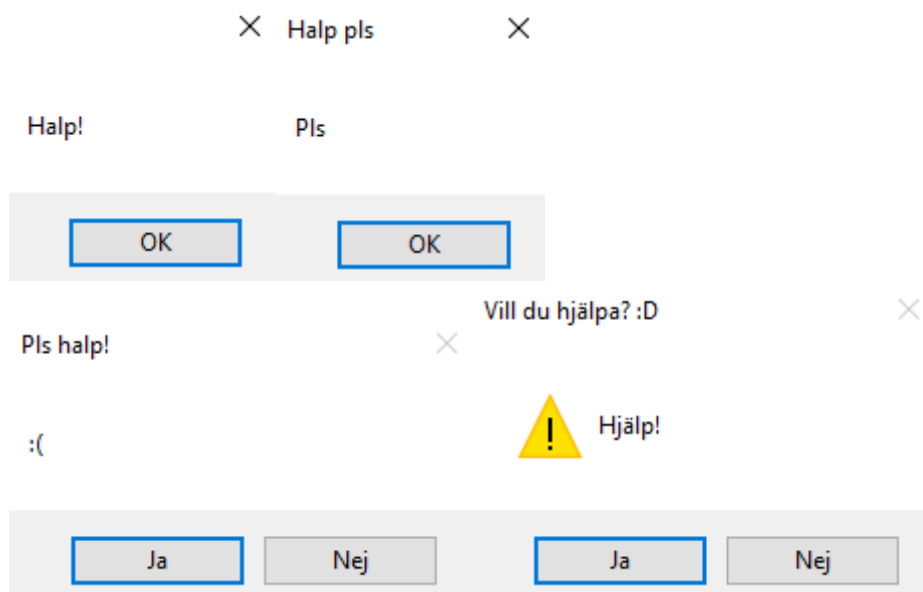
Exempel: `MessageBox.Show("You encountered an error!", "Error", MessageBoxButtons.OK, MessageBoxIcon.Error);`



Alla argument som gavs ovan finns det inget krav på att man måste använda sig av utan de är valfria. Exempel på hur vår kod ser ut för knappen i Form1:

```
// Öppnar MessageBox
private void btnMessage_Click(object sender, EventArgs e)
{
    // Text
    MessageBox.Show("Halp!");
    // Text + fönstertitel
    MessageBox.Show("Pls", "Halp pls");
    // Text + fönstertitel + Yes/No knappar
    MessageBox.Show(":(", "Pls halp!", MessageBoxButtons.YesNo);
    // Text+Titel+Knappar+Ikon
    MessageBox.Show("Hjälp!", "Vill du hjälpa? :D", MessageBoxButtons.YesNo, MessageBoxIcon.Warning);
}
```

Vilket ger oss 4 MessageBoxes efter varandra när vi trycker på knappen:



DialogResult

Med de tidigare MessageBoxes vi skapade använde vi oss av MessageBoxButtons.YesNo, men inget förutom att fönstret stänger sig händer när vi trycker på någon av knapparna(lägg märke till att vi inte kan trycka på kryss i de boxes vi definierat knappar för). Detta löser vi med DialogResult.

De olika typerna av knappar ger oss ett värde tillbaka beroende på vilken vi trycker på. Dessa värden kan vi fånga upp med DialogResult som vi sedan kan använda i if-satser och switchar.

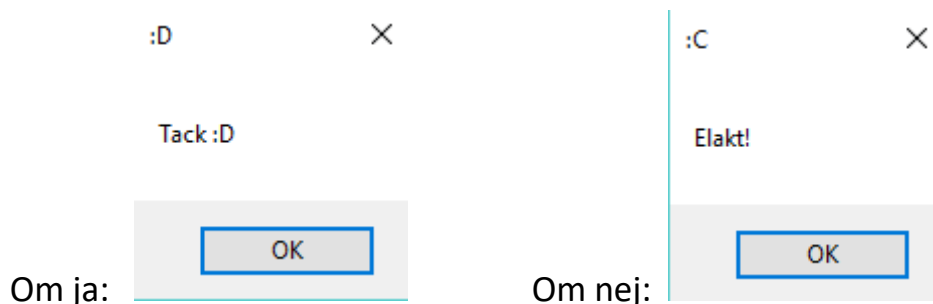
Vi ska nu använda oss av DialogResult på förra exemplet och använda oss av MessageBoxes för att skriva ut meddelanden till användaren beroende på vilket val som gjordes.

```
// Spara svaret i result
DialogResult result = MessageBox.Show("Hjälp mig!", "Hjälp :", MessageBoxButtons.YesNo,
    MessageBoxIcon.Question);

// Använd result i en switch och notifiera användaren
// beroende på val
switch (result)
{
    case DialogResult.Yes:
        MessageBox.Show("Tack :D", ":D");
        break;

    case DialogResult.No:
        MessageBox.Show("Elakt!", ":C");
        break;
}
```

Här skapar vi en DialogResult som lagrar svaret från vår MessageBox. Därefter kollar vi svaret i en switch-sats och skriver ut passande MessageBox. Resultat:



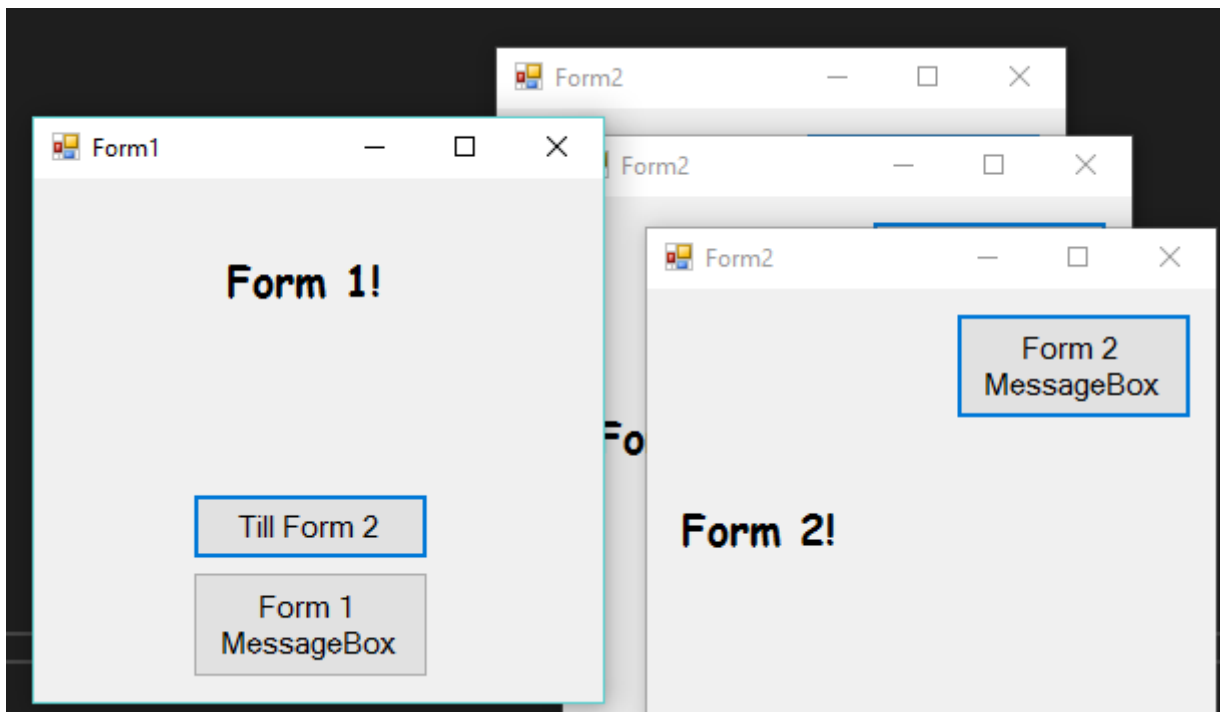
Öppna andra forms med kod

För att kunna öppna en ny form måste vi först skapa ett objekt av formen i fråga. Därefter, som med en MessageBox, behöver vi bara anropa metoden Show(). Exempel:

```
private void btnFrm2_Click(object sender, EventArgs e)
{
    // Skapa objekt av form2
    Form2 f2 = new Form2();

    // Öppnar en ny instans av Form2
    f2.Show();
}
```

Till skillnad från en MessageBox.Show() så tillåter denna kod oss att öppna flera instanser av Form2 vilket kan vara oönskvärt i vissa fall.



För att enkelt se till att vi bara kan öppna en instans av Form2 och inte kunna röra Form1 bör vi använda metoden ShowDialog(). Det denna metoden gör är att öppna formen som en dialog som hindrar oss från att röra den andra formen tills vi avslutat den dialogen.

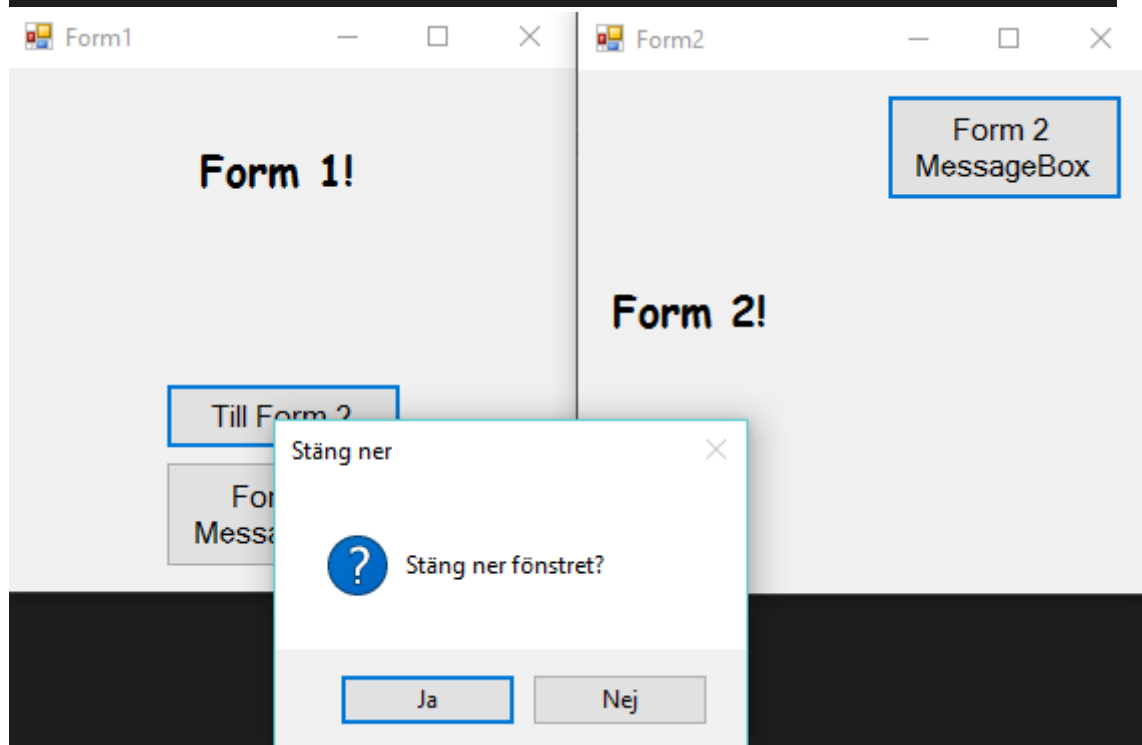
```
private void btnFrm2_Click(object sender, EventArgs e)
{
    // Skapa objekt av form2
    Form2 f2 = new Form2();

    // Öppnar en ny instans av Form2 som en dialog
    // dvs vi kan bara röra oss inom det fönstret
    f2.ShowDialog();
}
```

Till slut lägger vi till kod för vår MessageBox i Form2 där vi frågar om användaren vill stänga ner formen eller inte.

```
private void btnMessage_Click(object sender, EventArgs e)
{
    DialogResult result = MessageBox.Show("Stäng ner fönstret?", "Stäng ner",
        MessageBoxButtons.YesNo, MessageBoxIcon.Question);

    switch (result)
    {
        case DialogResult.Yes:
            Close(); // Stänger ner formen
            break;
        case DialogResult.No:
            MessageBox.Show("OK!");
            break;
    }
}
```



Vi kan använda oss av Forms för att skapa egna MessageBoxes och kombinera dessa med DialogResult. I egenskaper för knappar kan man välja att ge knappen ett värde under DialogResult fliken. Med hjälp av detta kan vi skapa våra egna MessageBoxes som exemplet med Microsoft Word.