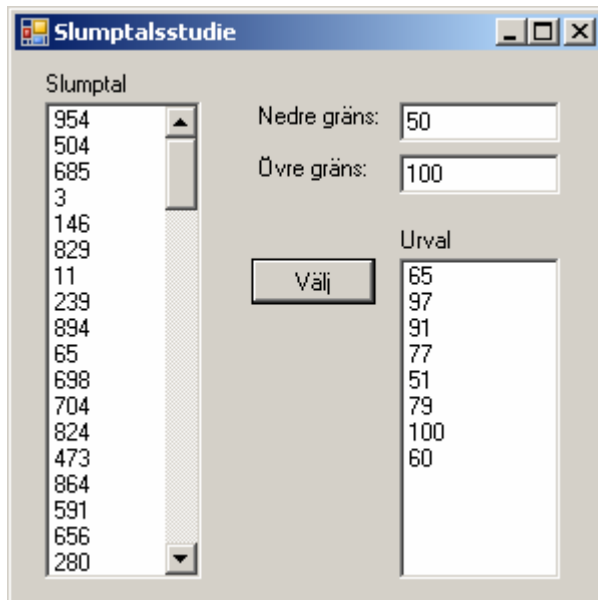


## Uppgifter – Windowsprogram 4

Gör gärna egna förändringar/förbättringar i programmen.

1. Här ska 100 slumpstal, mellan 1 och 1 000, skapas direkt då programmet startas (i formulärets *Load*-händelse alltså).

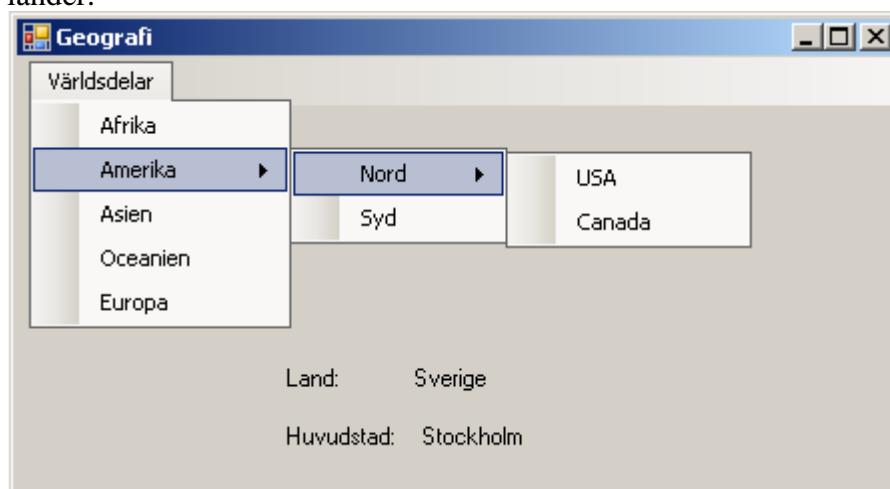


Urvalslistan ska, då ”Välj” klickats, bara innehålla talen i det intervall du bestämt.

[Tips](#)

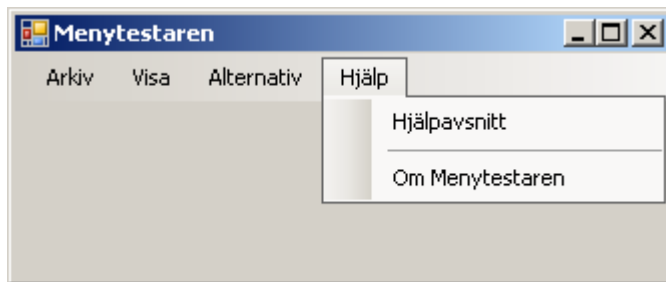
Extra: Om nedre gränsen lämnas tom så ta med alla tal från 1. Om övre gränsen lämnas tom så ta med alla tal upp till 1 000.

2. Presentera huvudstaden i det land du väljer. Det räcker om det fungerar för ett par, tre länder.



[Tips](#)

3. Skapa ett program med en menyrad som innehåller detta:



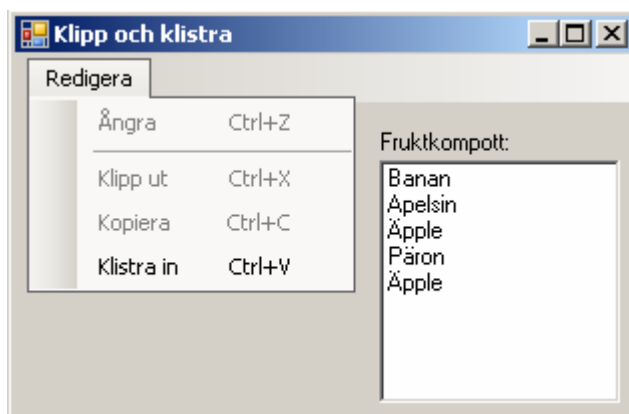
Övriga menyer kan du lämna tomma. Gör så att man kan växla mellan "Avancerad" och "Standard", det vill säga flytta på bockningen.

Då "Om Menytestaren" väljs, så ska en meddelanderuta visas:



#### Tips

4. Skapa ett formulär bestående av en meny och en lista som nedan:



Från början ska inget av menyalternativen vara aktiva (*Enabled*), och de fyra frukterna, banan, äpelsin, äpple och päron vara element i listan.

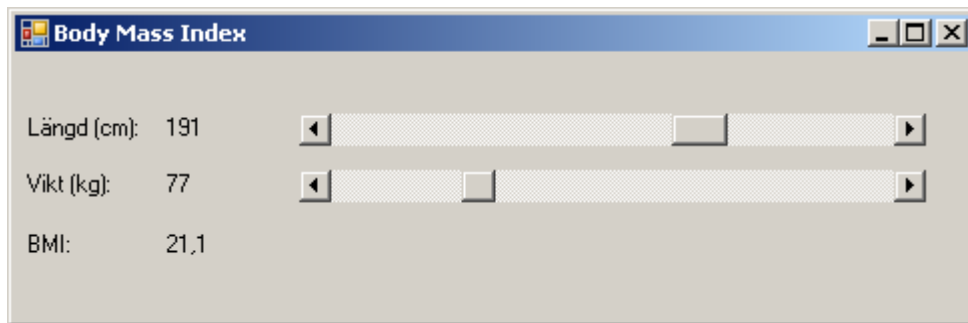
"Klipp ut" och "Kopiera" ska endast vara aktiva då en frukt är markerad och göra det

man kan förvänta sig av dem. "Klistra in" ska vara aktiv då något finns att klistra in, och ska lägga till den kopierade (eller utklippta) frukten sist i listan.

[Tips](#)

Extra: Få "Ångra"-alternativet att fungera!

5. Bygg ett bytt BMI-program, denna gång med rullningslister (*HScrollbar*):

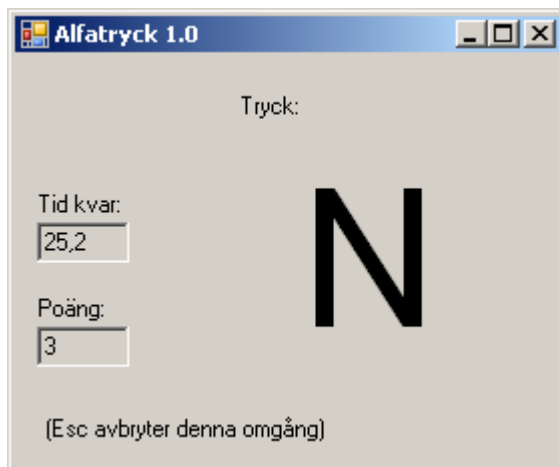


Ställ först in lämpliga värden för rullningslisternas egenskaper *Value*, *Minimum*, *Maximum*, *SmallChange* och *LargeChange*. Bestäm själv.

Värden bör visas direkt då programmet startas samt då rullningslisterna skrollas.

[Tips](#)

6. Konstruera ett litet spel som testar användarens reaktioner och tangentbordskännedom.



Spelidé: En slumpmässigt vald bokstav dyker upp då spelet startas. Samtidigt börjar tiden att räknas ned, från 30 till 0 sekunder, en tiondel i taget. Man skall då så snabbt som möjligt trycka rätt tangent och därmed få en poäng. Samtidigt kommer en ny bokstav fram och så fortsätter det till dess att tiden tar slut. Flest poäng vinner förstås.

Man ska kunna starta spelet med en *Enter*-tryckning och avbryta en omgång med *Esc*-tangenten.

[Tips](#)

7. I denna övning är det tänkt att du med hjälp av tre *PictureBox*-rutor och en *Timer* ska få en streckfigur att "springa" över skärmen.



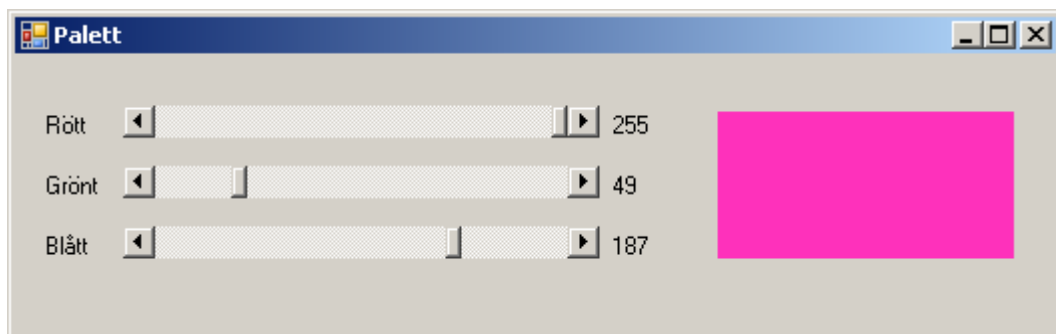
De två rutorna till höger, som visar kroppens olika rörelser, bör göras osynliga. Det enda som återstår nu är att låta en ruta omväxlande visa de två gubbarna och dessutom glida iväg till höger i bild!

Om du ändrar formulärets bakgrundsfärg (*BackColor*) till vit så smälter bilderna bättre in.

Knappen ska stanna löparen eller mer konkret sätta timerns egenskap *Enabled* till *false*.

[Tips](#)

8. Gör med hjälp av tre rullningslistor och en bildruta ett program där användaren kan blanda sig en egen färg.



För att åstadkomma detta skall metoden `Color.FromArgb` användas. Den tar tre argument som vardera ett värde på styrkan hos en grundfärg. Grundfärgerna är röd, grön och blå (därför `rgb`). Varje grundfärg kan ges ett värde mellan 0 och 255.

Exempel:

`Color.FromArgb(0, 0, 0)` ger svart färg

`Color.FromArgb(255, 255, 0)` ger gul färg

`Color.FromArgb(255, 255, 255)` ger vit färg

[Tips](#)

Tips:

1

Lägg in tal med en for-loop som denna:

```
for (int i = 0; i < 100; i++) // 100 varv
```

```
    listBox1.Items.Add(1+ rand.Next(1000)); // rand är ett Random-objekt
```

Välj genom att gå igenom alla listelement och testa:

```
for (int i = 0; i < 100; i++)
```

```
{
```

```
    tal = (int)listBox1.Items[i]; // Gör om listelementet till heltal med en cast
```

```
    if (tal >= nedre && tal <= övre)
```

```
        listBox2.Items.Add(tal);
```

```
}
```

2

Använd en *MenuStrip* (Sektionen *Menus & Toolbars*). Koden ska ligga på menyelementens klick-händelser.

3

Bockarna styrs av egenskapen *Checked*.

Avdelaren (strecket i Hjälp-menyn) skapas med ett bindestreck (-) i *Text*-egenskapen!

4

Kortkommandon (t ex CTRL+X) skapas med egenskapen *ShortcutKeys*.

Skapa en instansvariabel, minne, och spara det utklippta eller kopierade värdet där:

```
minne = listBox1.Text; // Spara den markerade frukten
```

Ta bort markeret element med raden:

```
listBox1.RemoveAt(listBox1.SelectedIndex);
```

5

Rullningslister (*HScrollBar* och *VScrollBar*) hittas i sektionen *All Windows Forms*.

*LargeChange* inträffar då man klickar bredvid handtaget. *SmallChange* vid klick på pilarna.

Avläs rullningslistens värde med egenskapen *Value*:

```
int längd = längdlist.Value;
```

6

Använd en *Timer*; från sektionen *Components*. Sätt intervallet till 100. I timerns *Tick*-händelse så räknas tiden ner och skrivs ut i etikett. Stanna timern då tiden == 0.

Övrig kod bör ligga i *Form1\_KeyDown*-metoden. Detta förutsätter att inget annat objekt har fokus. Använd därför etiketter eller sätt egenskapen *TabStop* till *false* i dina texrutor.

```
private void Form1_KeyDown(object sender, KeyEventArgs e)
{
    int kod = e.KeyValue;

    if (kod == 13) // Enter
    {
        timer1.Enabled = true; // Starta
    }
}
```

7

Kod för timerns *Tick*-händelse:

```
pictureBox1.Left += 10; // Öka avståndet från vänsterkanten
// Byt bild
```

Använd en instansvariabel för att hålla reda på vilken bild som visas för tillfället. Om du växlar mellan två bilder kan variabel vara en *bool* annars blir en *int* bäst.

8

Avläs de tre rullningslisternas tillstånd och spara i *int*-variabler:

```
int röd = rödlist.Value;
int grön = grönlist.Value;
int blå = blålist.Value;
```

Sätt sedan bakgrundsfärgen så här:

```
pictureBox1.BackColor = Color.FromArgb(röd, grön ,blå);
```