

PROGRAMMERING JAVA

DOCENDO

Innehållsförteckning

1 Allmänt om Java	5
I detta kapitel	5
Historia	5
Hur fungerar det att programmera?	6
Inspiration.....	9
Styrkan i Java.....	10
Övningsuppgifter.....	11
2 Utvecklingsverktyget.....	12
I detta kapitel	12
Utvecklingsverktyg.....	12
Säkerhetskopiera.....	12
Arbetsätt.....	12
Starta programmet	13
Arbetsytan.....	14
Skapa en ny applikation	15
Huvudmetoden	17
Kompilera och kör	18
Kompileringsfel.....	18
Snabbt om att stega i programmet.....	18
Stänga projekt.....	19
Öppna ett projekt.....	19
Stäng ännu en gång.....	20
Sammanfattning.....	20
Övningsuppgifter.....	21
3 Skriva egen programkod	22
I detta kapitel	22
Kommentarer	22
Slut på satsen	23
Data och algoritmer	23
Variabler	23
Enkel matte.....	24
Blanksteg	25
Skriftlägeskänslig	25
Skriva ut avslutningsinformation	26
Gör-det-själv	26
Övningsuppgifter.....	29
4 Flödesscheman och pseudokod	30
I detta kapitel	30
Vad är flödesscheman och pseudokod?.....	30
Vad är syftet med flödesscheman och pseudokod?	30
Flödesscheman.....	31
Pseudokod.....	32
Övningsuppgifter.....	32
5 Göra några enkla program.....	33
I detta kapitel	33
Du skriver ett komplett program.....	33
Skriva ut en sträng	38
Interaktivitet i Java	38
Ditt första interaktiva program.....	40
Länka till externa projekt.....	42
Fler variabeltyper	42
Mata in och presentera personuppgifter	45
Du gör en miniräknare	47
Övningsuppgifter.....	49
6 Mer om datatyper	51
I detta kapitel	51
Bitar och talsystem	51
Heltal	53
Flyttal	54
Sant eller falskt	54
Arrayer.....	55
Strängar och alfanumeriska tecken.....	56
Referenser	60
Aritmetiska operatorer	62
Omvandling mellan datatyper	65
Övningsuppgifter.....	66
7 Metoder	68
I detta kapitel	68
Varför metoder?.....	69
Fler metoder	69
Returnera flera värden via referenser	71
Kort om klasser, objekt och instanser	72
Lokala variabler	73
Några standardmetoder	74
Övningsuppgifter.....	78
8 De vanligaste nyckelorden	80
I detta kapitel	80
Loopar.....	80
Andra vanliga nyckelord	83
Mer om jämförande uttryck.....	84
Övningsuppgifter.....	86
9 Större projekt	87

I detta kapitel	87
Mycket källkod blir dålig soppa	87
Övningsuppgifter.....	89

10 Grafisk programmering90

I detta kapitel	90
Bildpunkter	90
RGB	90
Grafiska applikationer.....	91
Du gör en grafisk applikation	92
Övningsuppgifter.....	94

11 Skriv en fraktalzoomare 95

I detta kapitel	95
Fraktalteori	95
Grundplåten	96
Mandelbrot.....	97
Övningsuppgifter.....	102

12 Gå vidare105

I detta kapitel	105
Programmering	105
Medföljande övningsexempel	105
Övningsfilen FractalZoom	105
Övningsfilen FractalClick	106
Övningsfilen Sort.....	106
Övningsfilen Starfield	106
Övningsfilen Worms	106
Övningsfilen Sierp.....	106

13 Nyckelord i Java.....107

De vanligaste nyckelorden	108
---------------------------------	-----

14 Teckentabell115

15 Facit.....120

Sakregister 135

Starta programmet

Du ska börja med att starta programmet (utvecklingsverktyget).

1. Klicka på **Start**-knappen, välj **Alla Program, NetBeans**, välj **NetBeans IDE 6.8**.

När programmet startas för allra första gången kan det se annorlunda ut mot hur det ser ut då du startar det fortsättningsvis.



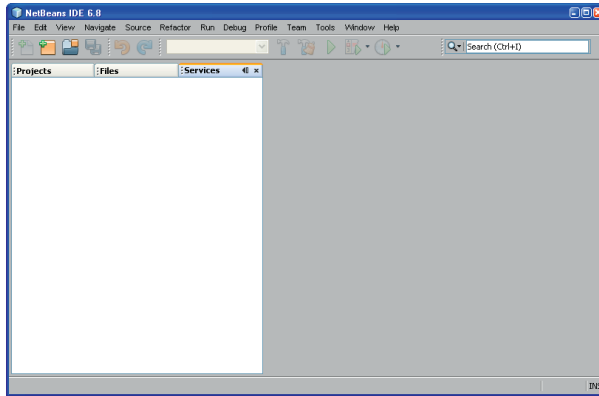
Ser programmet annorlunda ut kan du hoppa över nästföljande steg och gå direkt till avsnittet Arbetsytan.

2. Klicka på **Stäng**-knappen på startfliken.



Arbetsytan

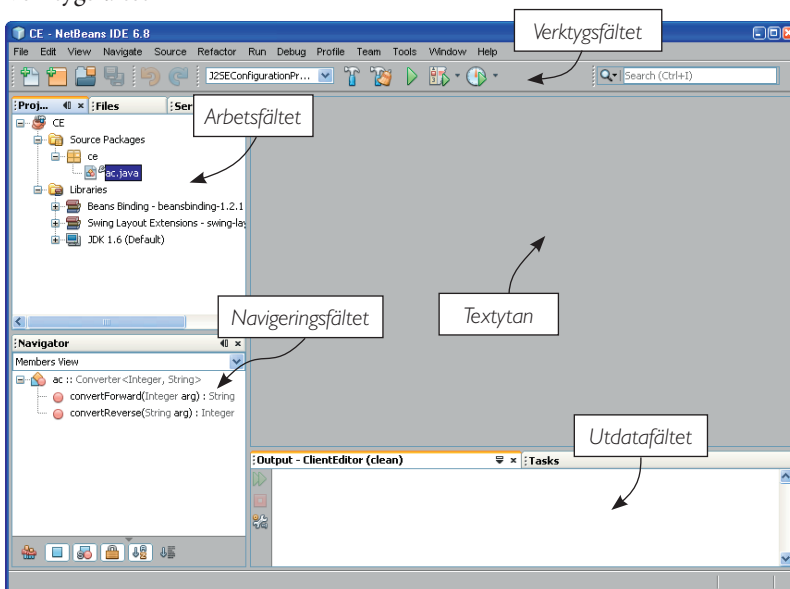
Du ser nu en tom arbetsytan:



Benämningar

De vanligaste delarna du kommer att använda dig av är:

- Textytan
- Arbetsfältet
- Navigeringsfältet
- Utdatafältet
- Verktygsfältet



Textytan

Här kommer den faktiska texten i källkoden synas när du programmerar.

Arbetsfältet

Ett projekt kan innehålla ett stort antal olika filer. Alla filer i ett projekt kommer att listas i Arbetsfältet. Om du har flera projekt öppna samtidigt, så hittar du samtliga i arbetsfältet.

Navigeringsfältet

Detta fält dyker upp när ett projekt är valt. Håri får du överskådlig information om den aktiva källkodsfilen och du kan dubbelklicka på element för att direkt visa den delen av källkodsfilen, vilket är praktiskt när du arbetar med stora filer.

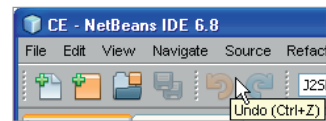
Utdatafältet

Även Utdatafältet visas när ett projekt är öppet. I Utdatafältet visas textinformation till dig som användare. Här visas kompilersningsinformation när du kompilerar och textutskrifter när du avlusar. Notera att detta endast syns när man har öppnat ett projekt och kört resultatet. När man kompilerar växlar NetBeans automatiskt till "Build"-utskrifter och när man kör ett program så visas i stället information för avlusning.

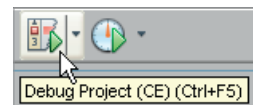
Build, eller att *bygga*, kallar NetBeans processen att kompilera och länka ett eller flera projekt till en körbar fil. Ett projekt kan ju innehålla flera individuella källkodsfiler.

Verktygsfältet

På verktygsfältet hittar du knappar som du säkert känner igen till namnet: **Save all files** och **Undo**.



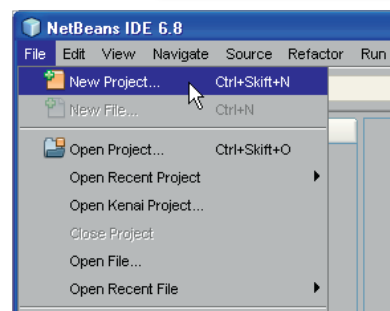
Men det finns även knappar som du måhända inte har sett förut, som till exempel **Debug Project**.



Skapa en ny applikation

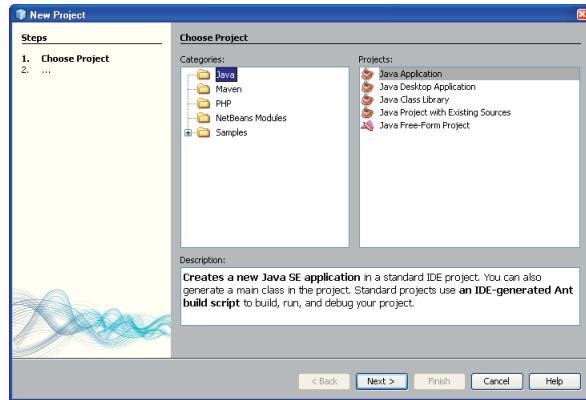
Nu ska du skapa ditt första program. Det finns färdiga funktioner i NetBeans för att skapa ett tomt program som inte gör någonting.

1. Gå in under menyn **File**, **New Project...**



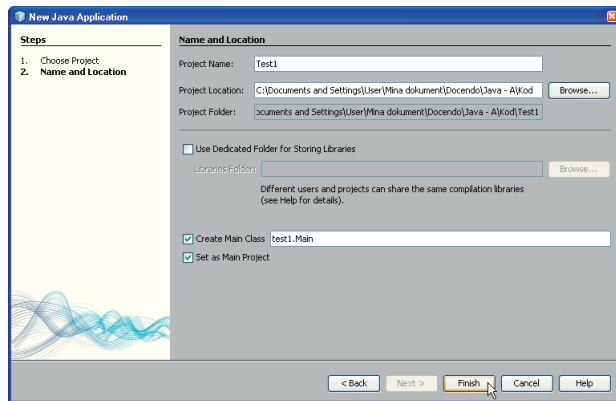
Nu visas en dialogruta, se nedan.

2. Välj **Java** i mitten (kategorikolumnen) och sedan **Java Application** till höger (projektkolumnen). Klicka på knappen **Next**.
3. Fältet **Project Name** är nu markerat; skriv **Test1**.
4. Klicka på knappen **Finish**.



Projektet är skapat och klart.

Källkodsfiler i Java har filtillägget **.java**. Nu har du även skapat projektfiler för NetBeans som håller reda på innehållet i dina projekt (de viktigaste ligger i katalogen **nbproject** och heter **project.properties** samt **project.xml**).



I Textytan visas nu källkoden till ditt första program: **Main.java**.

```

1  |  /*
2  |  * To change this template, choose Tools | Templates
3  |  * and open the template in the editor.
4  |  */
5  |
6  |  package test1;
7  |
8  |  /**
9  |   *
10 |   * @author User
11 |   */
12 |  public class Main {
13 |
14 |      /**
15 |       * @param args the command line arguments
16 |       */
17 |      public static void main(String[] args) {
18 |          // TODO code application logic here
19 |      }
20 |
21 |  }
22 |

```

- Alla rader med grå text är kommentarer (rader 1–4, 8–11, 14–16, 18).
- Rad 6 anger vilket ”paket” källkoden befinner sig i. (Paketering av källkod är ett sätt att visa vilka delar som hör ihop.) Ignorera denna rad tills vidare.
- På rad 12 står det vilken klass du jobbar i. Klassen har samma namn som filen det står i; i övrigt kan du ignorera även den här raden tills vidare.
- På raderna 17 och 19 står själva programkoden.

Alla ord i blått är nyckelord för Java. Det betyder att de är reserverade för en specifik tolkning av kompilatorn.

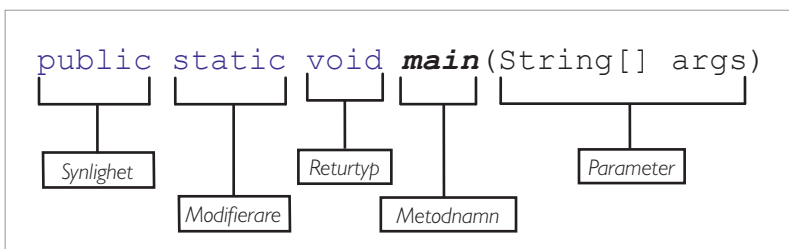
Dessa färger är standard i NetBeans, men om du använder andra verktyg så kanske du har andra färger. Det går också att ändra inställningarna om du vill ha en annan färgsättning.

Huvudmetoden

main

Det är **main** som är huvudmetoden i en Java-applikation; det vill säga på den raden börjar exekveringen av ditt program. När sista raden i **main** exekverats så avslutas programmet. **main** har samma karakteristika som alla andra metoder:

- En synlighet.
- Noll, en eller flera modifierare.
- En returtyp.
- Ett namn.
- Noll, en eller flera parametrar avskilda med kommatecken och omgärdade med parenteser.
- Klammerparenteser som anger innehållet i metoden.



Du ska få allt detta förklarat grundligt för dig i senare kapitel. Tills vidare nöjer vi oss med att säga att all källkod som du skriver in i **main** (efter vänster klammerparentes men före höger klammerparentes) kommer att exekveras när du kör programmet.

När det kommer till vänsterklammerparentesens placering så gäller att den kan placeras hur som helst mellan högerparentesen (i **main**) och den första satsen.

Kompilera och kör

Kommandot Kör

Nu ska du testköra applikationen.

1. Gå in under menyn **Run**, välj **Run Main Project** (kortkommando **F6**).
2. Ditt program startas, men det avslutas lika fort, eftersom du inte ännu lagt till något innehåll.

Kompileringsfel

Felaktig källkod ger upphov till kompileringsfel. Kompileringsfelen är lätta att upptäcka eftersom de gör att du inte kan kompilera och köra applikationen. De visas även med en röd understrykning i Textytan. Du ska nu testa att orsaka ett kompileringsfel.

1. Lägg till tecknet **a** i början på parametern **String[] args** så att det i stället står **aString[] args**.

Direkt dyker en röd understrykning upp under texten **aString**, och om du håller pekaren över så visas en inforuta med mer information.

```

14 /**
15  * @param args the command l
16  */
17 public static void main(aString[] args) {
18     // TODO code application logic here
19 }
20

```

Först i inforutan står felets typ, i detta fall att kompilatorn inte hittar en viss symbol. På andra raden står symbolens namn, det vill säga **aString**. På sista raden står var felet befinner sig (det är överflödigt information i detta läget eftersom vi redan tydligt ser rödmarkeringen).

2. Korrigera felet genom att ta bort **a:**et så att det återigen står **String[] args**.

Snabbt om att stega i programmet

För att ge dig en bild av hur programexekveringen ser ut, så ska du nu prova att stega igenom programmet. Med ”stega i programmet” menas att du exekverar programmet rad för rad. Varje ny rad körs när du trycker på ”stega-knappen”.

1. Tryck kortkommandot **F7**.

Programmet laddas och exekveringen börjar på den första och sista raden i programmet (eftersom raden bara innehåller en rad). Sedan pausas exekveringen och NetBeans väntar på att du ska fortsätta exekveringen. Det ska se ut så här:

```

17 | public static void main(String[] args) {
18 |     // TODO code application logic here
   | }
20 |

```

Den gröna pilen visar vilken rad exekveringen är på för tillfället.

2. Tryck **F7** igen.

Programmet exekveras färdigt (eftersom det inte finns någon fler kodrad i programmet) och avslutas därför; efteråt blir utdatafältet sig likt igen.

Stänga projekt

1. Stäng **Test1.cpp**.

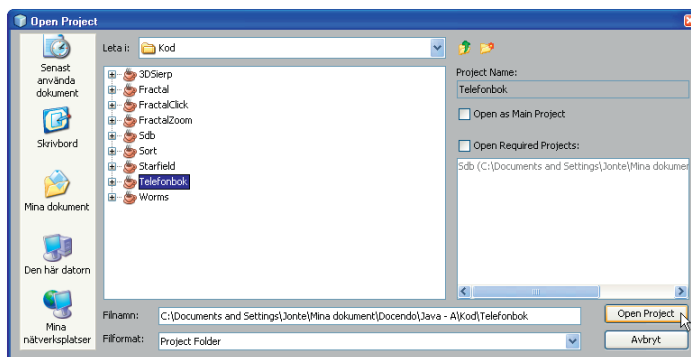
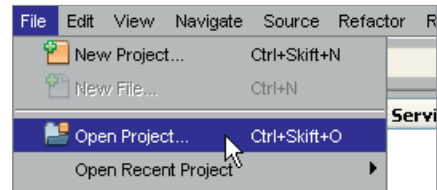
2. Gå in under menyn **File**, välj **Close Project (Test1)**.

Öppna ett projekt

Du ska nu öppna ett färdigt projekt som följer med kursen och titta lite på källkoden däri.

1. Välj **File**, **Open Project...**

- ### 2. Bläddra till den plats där projektfilerna som medföljer kursen finns. Klicka på **Telefonbok** och klicka sedan på knappen **Open Project**.



Övningsuppgifter

- Övning 2.1 Vad är orsakerna till att du som programmerare kommer att använda mycket kortkommandon i stället för att arbeta med musen?
- Övning 2.2 Vissa delar av källkoden får grå färg i NetBeans, vilka?
- Övning 2.3 Vad betyder ord i källkoden med blå färg?
- Övning 2.4 Vilket filtillägg används för källkodsfiler?
- Övning 2.5 Vad heter huvudmetoden i ett program?
- Övning 2.6 Genomför följande procedur:
1. Skapa en ny, enkel applikation och kalla den för **MittTest1**.
 2. Använd kortkommandot för att exekvera.
 3. Öppna källkodsfilen.
 4. Exekvera igen.
 5. Avsluta NetBeans (**File, Exit**).