



Objects First With Java
A Practical Introduction Using BlueJ

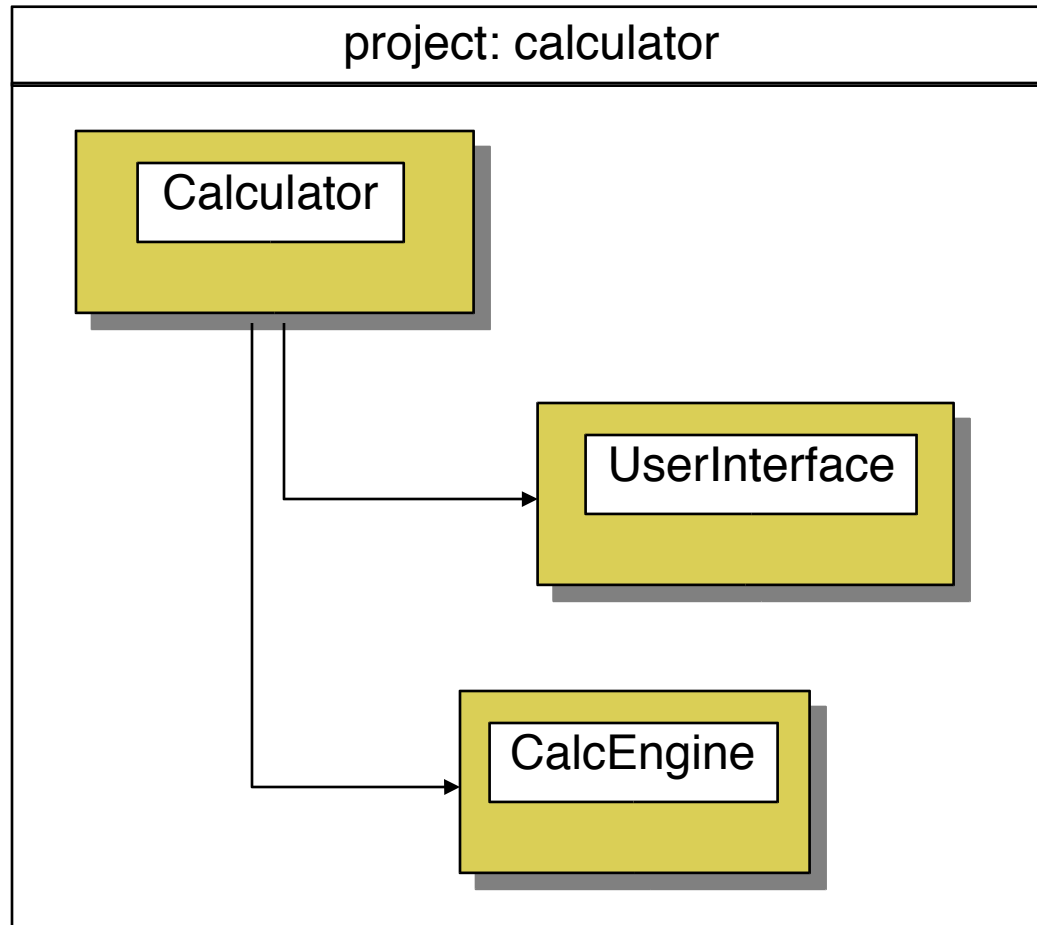
Att använda Java utan(för) BlueJ



BlueJ projekt

- Ett BlueJ projekt lagras i en katalog på disken.
- Ett BlueJ projekt lagras i ett antal filer.
- Några filer innehåller källkoden, några den kompillerade koden, några till lagrar annan information.
- BlueJ använder standard Java-formatet för vissa filer och sedan utökar informationen i övriga filer i paketet.

BlueJ:s katalogstruktur



c:\bluej\calculator\
bluej.pkg
bluej.pkh
Calculator.java
Calculator.class
Calculator.ctxt
UserInterface.java
UserInterface.class
UserInterface.ctxt
CalcEngine.java
CalcEngine.class
CalcEngine.ctxt

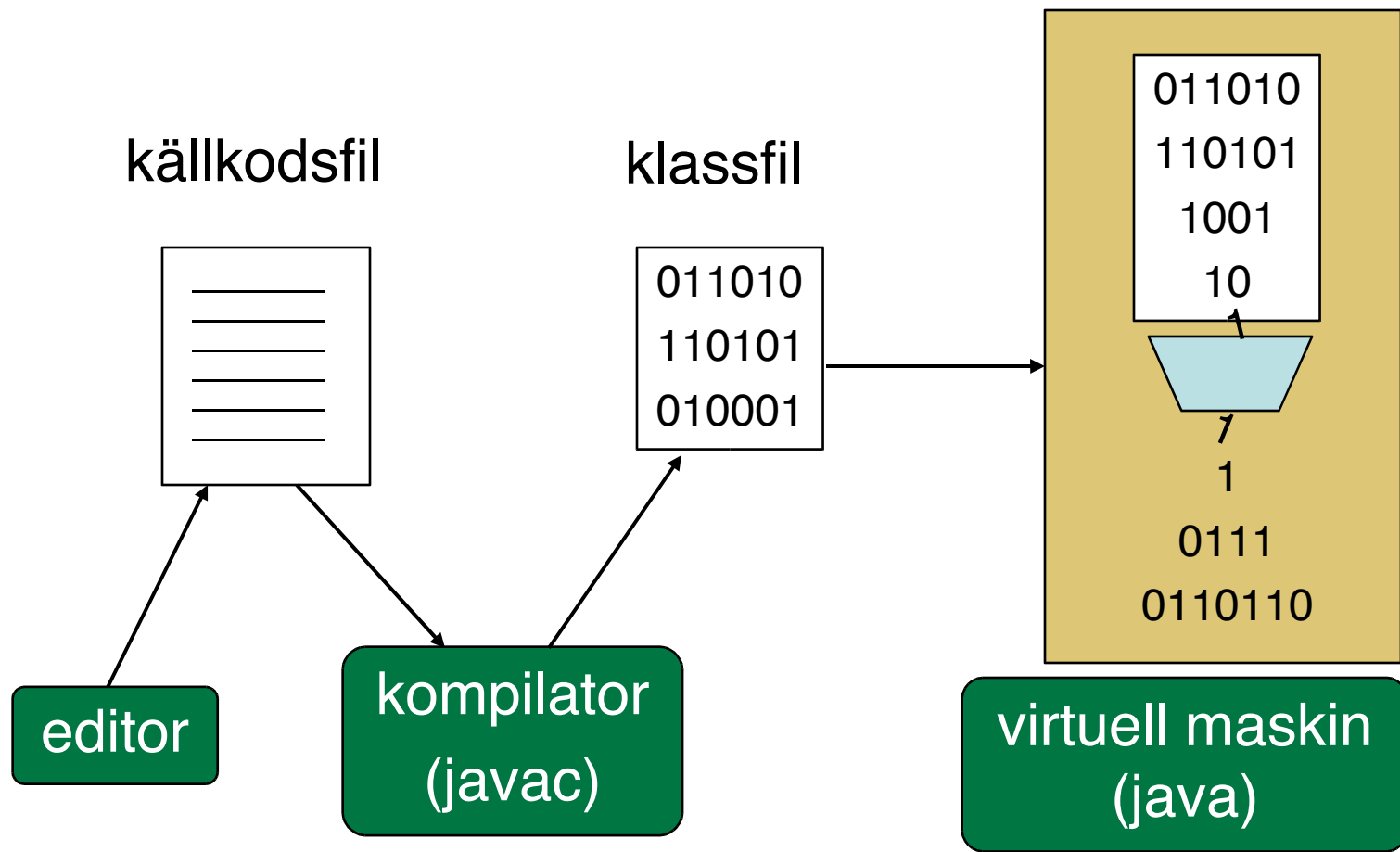
BlueJ:s filer

- bluej.pkg - paketfil. Innehåller information om klasserna i paketet. En per paket.
- bluej.pkh - backup av paketfilen.
- *.java - Java källkod (text). En per klass.
- *.class - Java bytekod. En per klass.
- *.ctxt - BlueJ kontextfil. Innehåller BlueJ-information om filen. En per klass.

Standard Javafiler

- **källkodsfiler: *.java**
Innehåller ett Javaprogram i textform.
Skrivs in av programmeraren.
- **klassfiler: *.class**
Innehåller ett Javaprogram i
bytekodsform. Skrivs in av
programmeraren.

editera-kompilera-exekvera



Editering

- En fil editeras av någon editor
 - textedit, emacs, vi, ...
- Se upp med ordbehandlare som t.ex. Word: de sparar standardmässigt inte filer i enkelt textformat.
- Det gäller också att komma ihåg att spara filen efter varje ändring.

JDK -Java Development Kit

- Kompilering av och exekvering av Javaprogram görs via kommandon i ett skalfönster
- Microsoftsystem: DOS shell
- Unix: Unix shell
- Man måste se till att de program som körs som ett resultat av kommandona kommer att hittas av systemet.

Kompilering

- JDK-kompilator: `javac`
- Man skriver:
`javac <filnamn>`
- Det kompilerar källkodsfilen `<filnamn>` och alla klasser som beror på den
- Exempel:
`cd C:\bluej\zuul`
`javac Game.java`

Felmeddelande

```
C:\bluej\zuul> javac Game.java
Game.java:22: ';' expected.
    private Parser parser
                        ^
1 error
C:\bluej\zuul>
```

Innebär att programmeraren måste öppna källkodsfilen i editorn, leta upp rad 22, fixa felet, spara(!) filen och kompilera igen.

The Java Compiler - javac



```
Microsoft Windows 2000 [Version 5.00.2195]
(C) Copyright 1985-2000 Microsoft Corp.

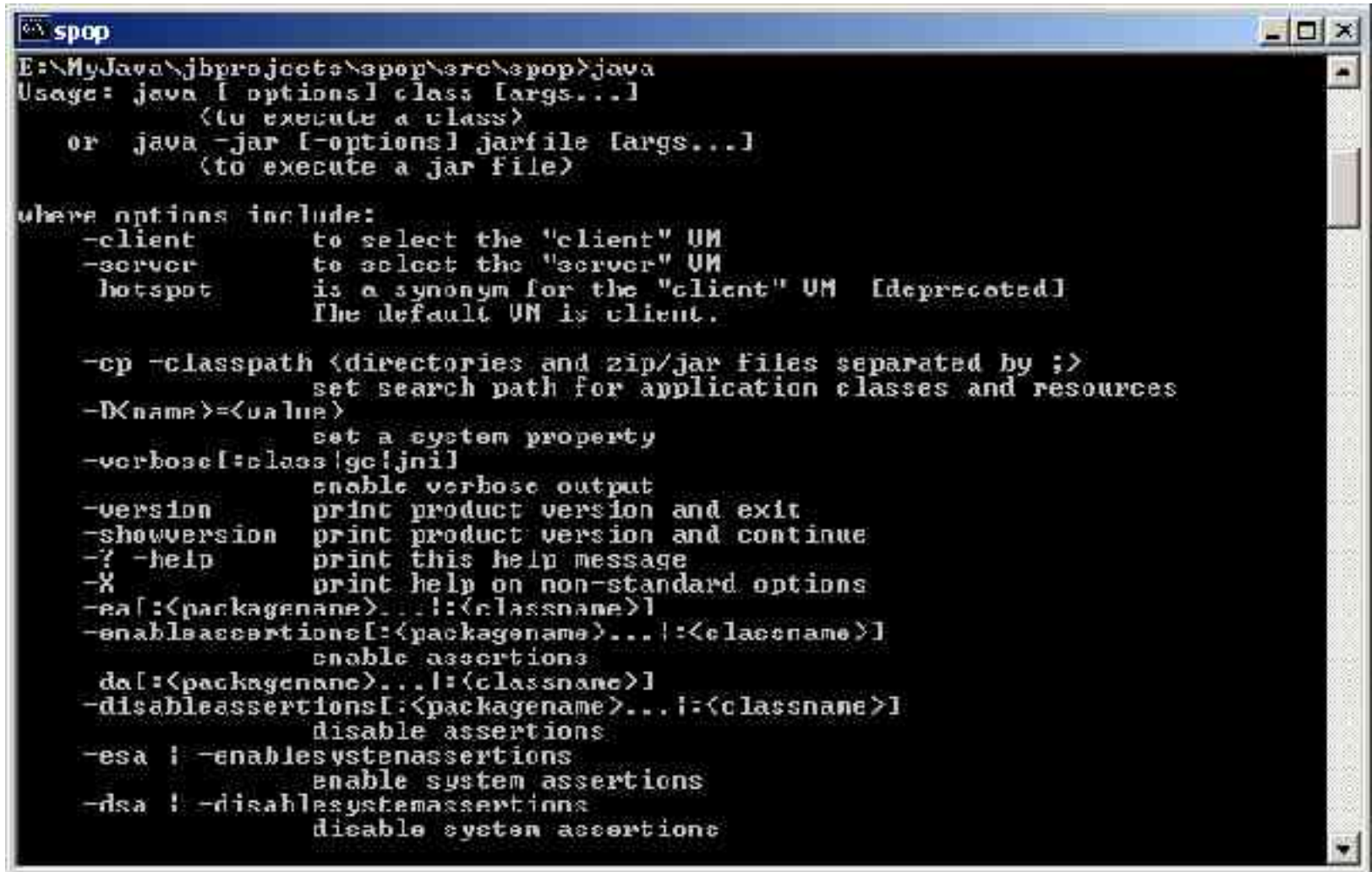
E:\MyJava\jbprojects\spop\src\spop>javac
Usage: javac <options> <source files>
where possible options include:
    -g                      Generate all debugging info
    -g:none                 Generate no debugging info
    -g:<lines,vars,source>  Generate only some debugging info
    -nowarn                 Generate no warnings
    -verbose               Output messages about what the compiler is doing
    -deprecation            Output source locations where deprecated APIs are used
    -classpath <path>      Specify where to find user class files
    -sourcepath <path>     Specify where to find input source files
    -bootclasspath <path>  Override location of bootstrap class files
    -extdirs <dirs>        Override location of installed extensions
    -d <directory>         Specify where to place generated class files
    -encoding <encoding>   Specify character encoding used by source files
    -source <release>       Provide source compatibility with specified release
    -target <release>       Generate class files for specific VM version
    -help                  Print a synopsis of standard options

E:\MyJava\jbprojects\spop\src\spop>
```

Exekvering

- `C:\bluej\zuul> java Game`
- “java” startar den virtuella maskinen (JVM).
- Den namngivna klassen laddas av JVM och exekveringen startar.
- Andra klasser laddas när de behövs.
- Förutsätter att filen `Game.java` finns i bytekodsform i en katalog som ingår i aktuell sökväg.

The Java Interpreter - java



```
spop
E:\MyJava\jbp\projects\spop\src\spop>java
Usage: java [-options] class [args...]
           (to execute a class)
    or java -jar [-options] jarfile [args...]
           (to execute a jar file)

where options include:
    -client          to select the "client" VM
    -server          to select the "server" VM
                     hotspot is a synonym for the "client" VM [deprecated]
                     The default VM is client.

    -cp -classpath <directories and zip/jar files separated by ;>
                     set search path for application classes and resources
    -D<name>=<value> set a system property
    -verbose[:class[:gc[:jni]]
                     enable verbose output
    -version          print product version and exit
    -showversion      print product version and continue
    -? -help          print this help message
    -X                print help on non-standard options
    -ea[:<packagename>...[:<classname>]]
    -enableassertions[:<packagename>...[:<classname>]]
                     enable assertions
    -da[:<packagename>...[:<classname>]]
    -disableassertions[:<packagename>...[:<classname>]]
                     disable assertions
    -esa ! -enablesystemassertions
                     enable system assertions
    -dsa ! -disablesystemassertions
                     disable system assertions
```

Problem: Exekvera vad?

- Om vi provar:

```
C:\bluej\zuul> java Game  
Exception in thread "main"  
java.lang.NoSuchMethodError: main
```

- Problem: Hur skall JVM veta var/hur exekveringen skall starta?

main-metoden

- Lösning: Javasystemet börjar alltid med att exekvera en metod som har en fastställd signatur:

```
public static void main(String[] args)
{
    ...
}
```

- Det måste alltså finnas en sådan metod i den klass som anges på kommandoraden!

main-metoden (2)

- “main” måste finnas
- “main” måste vara public
- “main” måste deklareras static (klass metod)
- “main” måste ha en String array som parameter
- Bara “main” definierad på detta sätt kan startas av JVM.

```
public static void main(String[] args)
{
    ...
}
```


Main method - exempel

```
public static void main(String[] args)
{
    Game game = new Game();
    game.play();
}
```

- Idealiskt bör main-metoden
 - skapa ett objekt;
 - anropa en metod i objektet.

Paket

Klassbiblioteken i Java är organiserad i paket (*packages*).

Man kan skapa egna paket.



Egendefinierade paket

```
package graphics;
```

```
class Circle  
{  
    ...  
}
```

```
package graphics;
```

```
class Rectangle  
{  
    ...  
}
```

Egendefinierade paket

- Skapa en katalog med samma namn som namnet på paketet (*graphics*)
- Lägg de kompilerade filerna (*.class) i denna katalog.
- Se till att katalogen kan hittas via systemets CLASSPATH
- Alternativt: Lägg *graphics* i aktuell katalog.

CLASSPATH

UNIX:

```
setenv $CLASSPATH .:~/classes
```

DOS shell (Windows):

```
set Path=.;~/classes
```

Kompilatorn kommer nu att hitta katalogen *graphics* om den är en underkatalog i *classes*.

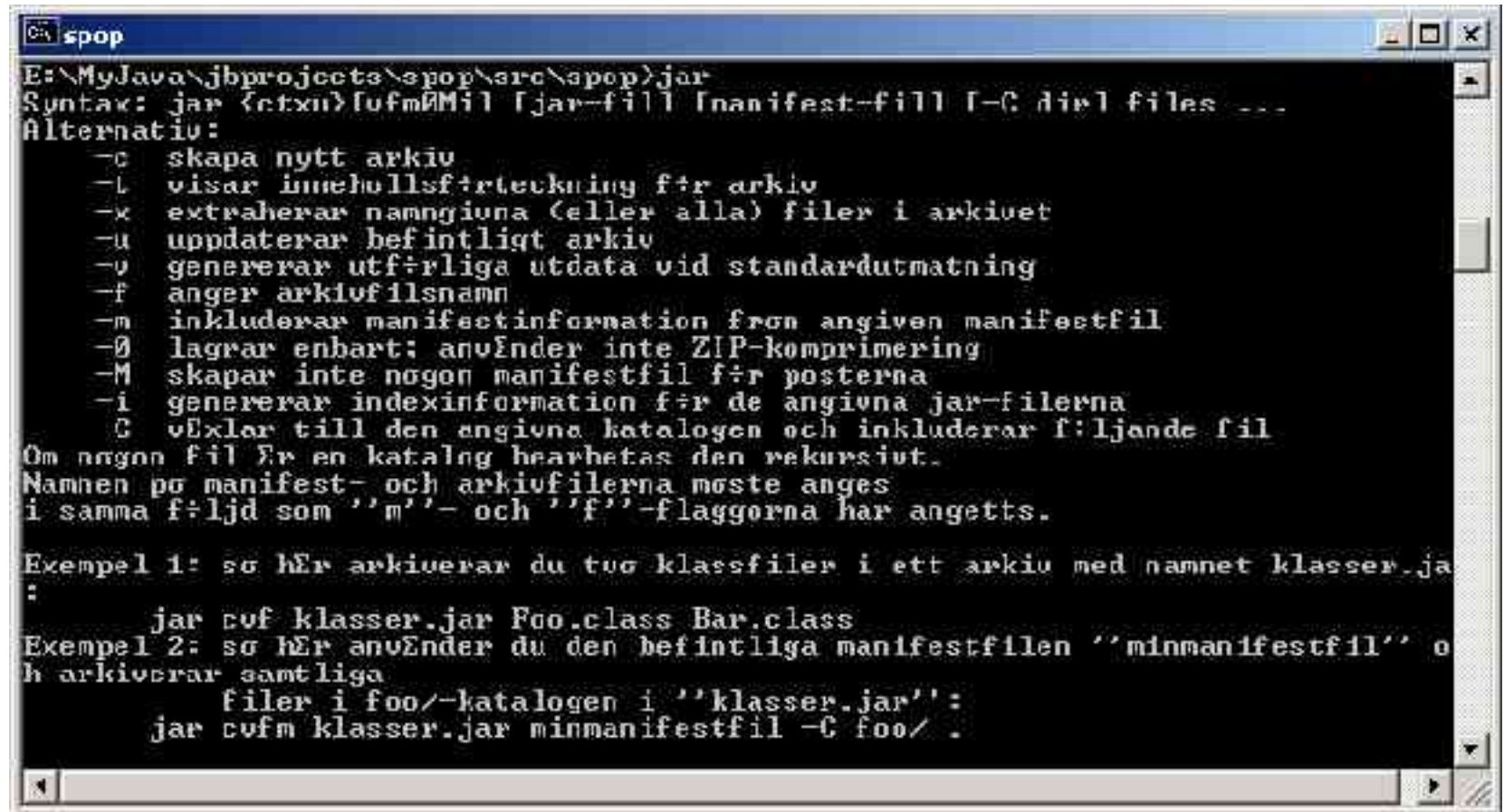
Java Archive Tool

Ett verktyg för att skapa och hantera Java Archive (JAR) Files.

En JAR-fil är en ZIP-fil som innehåller Java-klasser.



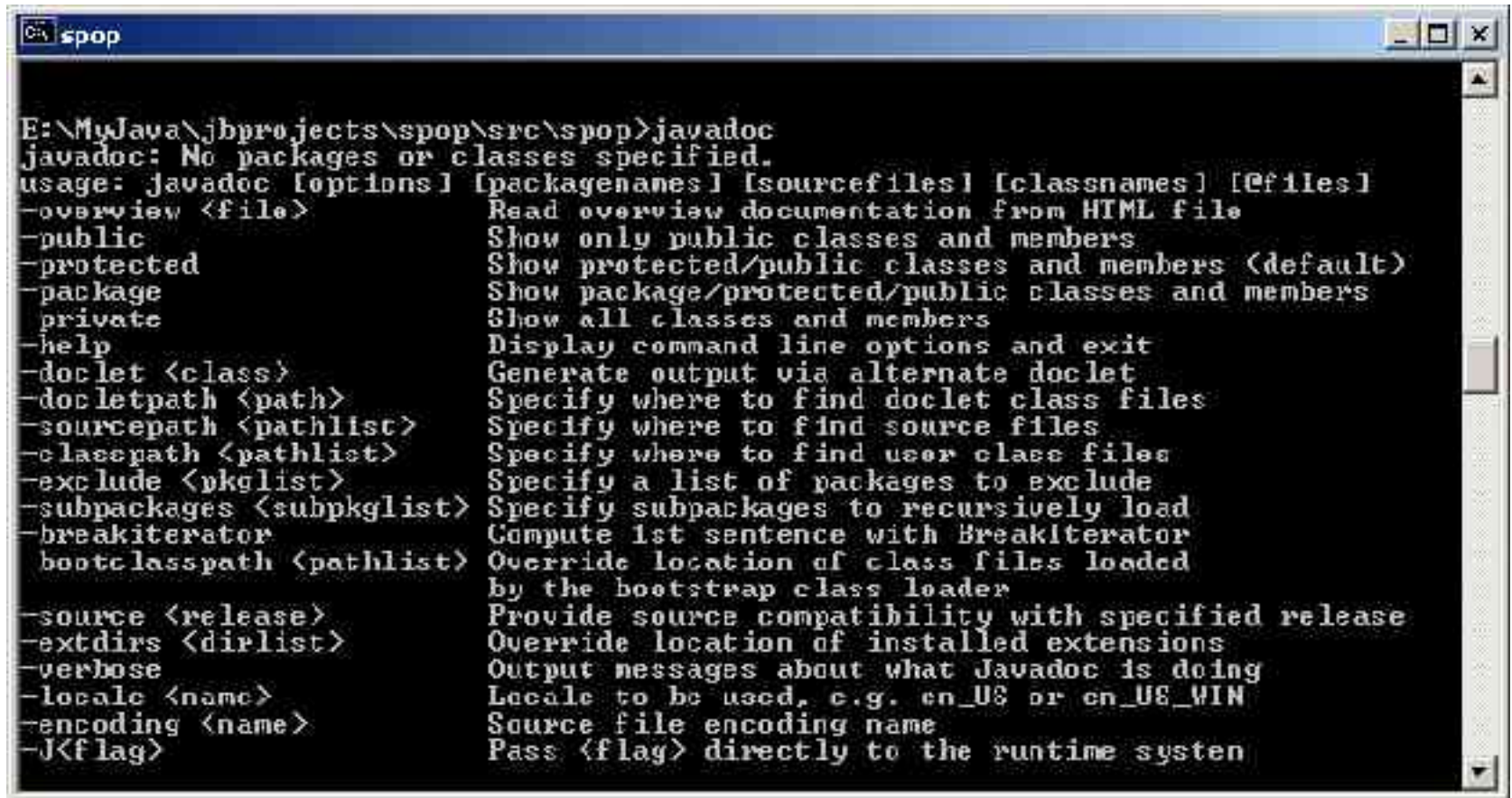
Java Archive Tool - jar



```
E:\MyJava\jbprojects\spop\src\spop>jar
Syntax: jar {ctxu}ufmMfil [jar-fill] [manifest-fill] [-C dir] files ...
Alternativ:
  -c skapa nytt arkiv
  -l visar innehållsförteckning för arkiv
  -x extraherar namngivna (eller alla) filer i arkivet
  -u uppdaterar befintligt arkiv
  -v genererar utförliga utdata vid standardutmatning
  -f anger arkivfilnamn
  -m inkluderar manifestinformation från angiven manifestfil
  -0 lagrar enbart; använder inte ZIP-komprimering
  -M skapar inte någon manifestfil för posterna
  -i genererar indexinformation för de angivna jar-filerna
  -C växlar till den angivna katalogen och inkluderar följande fil
Om någon fil är en katalog bearbetas den rekursivt.
Namnen på manifest- och arkivfilerna måste anges
i samma följd som ''m''- och ''f''-flaggorna har angetts.

Exempel 1: så här arkiverar du två klassfiler i ett arkiv med namnet klasser.jar
:
jar cvf klasser.jar Foo.class Bar.class
Exempel 2: så här använder du den befintliga manifestfilen ''minmanifestfil'' o
h arkiverar samtliga
filer i foo/-katalogen i ''klasser.jar':
jar cvfm klasser.jar minmanifestfil -C foo/ .
```


Java Documentation Generator



```
E:\MyJava\jbprojects\spop\src\spop>javadoc
javadoc: No packages or classes specified.
usage: javadoc [options] [packagenames] [sourcefiles] [classnames] [@files]
-overview <file>          Read overview documentation from HTML file
-public                   Show only public classes and members
-protected                Show protected/public classes and members (default)
-package                  Show package/protected/public classes and members
-private                  Show all classes and members
-help                     Display command line options and exit
-doclet <class>           Generate output via alternate doclet
-docletpath <path>        Specify where to find doclet class files
-sourcepath <pathlist>    Specify where to find source files
-classpath <pathlist>     Specify where to find user class files
-exclude <pkglist>        Specify a list of packages to exclude
-subpackages <subpkglist> Specify subpackages to recursively load
-breakiterator             Compute 1st sentence with Breakiterator
-bootclasspath <pathlist> Override location of class files loaded
                           by the bootstrap class loader
-source <release>         Provide source compatibility with specified release
-extdirs <dirlist>        Override location of installed extensions
-verbose                  Output messages about what Javadoc is doing
-locale <name>            Locale to be used, e.g. en_US or en_US_WIN
-encoding <name>          Source file encoding name
-J<flag>                  Pass <flag> directly to the runtime system
```




JBuilder

Kraftfullt utvecklingsverktyg som kan laddas ner från

<http://www.borland.com/jbuilder>

Enterprise

Developer

Foundation

JBuilder versions

Enterprise

JBuilder 2005 Enterprise speeds Enterprise JavaBeans,™ Web, Web Services, XML, mobile, and database application development with two-way visual designers and rapid deployment to leading J2EE™ application servers.

Developer

JBuilder 2005 Developer is a powerful Java IDE that speeds development of Web, database, and mobile applications — now with JavaServer Faces support and an integrated profiler

Foundation

JBuilder 2005 Foundation is a free Java IDE that speeds development with an integrated editor, debugger, compiler, visual designers, wizards, refactorings, code formatting, and much more.

Testning utan BlueJ

- Testprogram måste skrivas separat;
- Alla anrop av testmetoder måste skrivas in och samlas i en eller flera testmetoder;
- Om tester beror på resultatet av tidigare tester måste testprogrammet editeras och kompileras igen;
- Testprogrammet måste explicit skapa de objekt som skall testas.