

Using Java without BlueJ

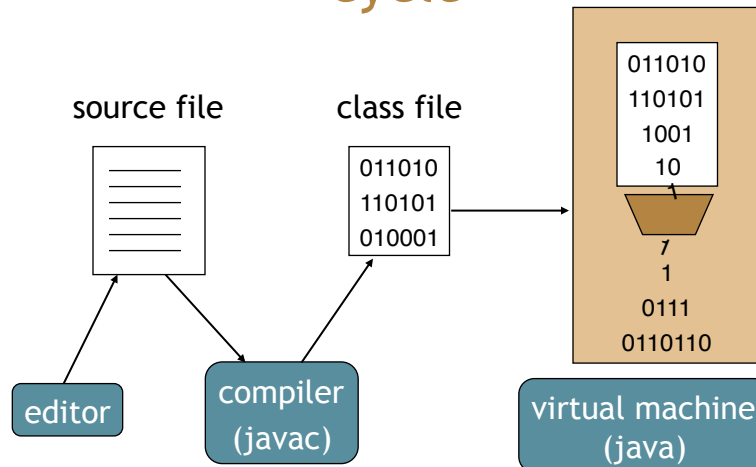
4.0

Standard Java files

- **source files:** `*.java`
Java source files contain the source code in readable form, as typed in by the programmer.

class files: `*.class`
Java class files contain byte code (a machine readable version of the class). They are generated by the compiler from the source file.

The edit-compile-execute cycle



Editing

- A file can be edited in any text editor
 - Notepad, emacs, jEdit, PFE, vi, ...
- Don't use Word: by default, Word does not save in text format
- Make sure to save with a .java filename before compiling!

Command line invocation

- Compilation and execution of Java in JDK are done from a command line
- On Microsoft systems: DOS shell
- On Unix: Unix shell
- Must make sure that the commands for compiler and runtime are in the command path.

Compiling

- Name of the JDK compiler: `javac`
- To invoke:
`javac <source name>`
- compiles `<source name>` and all classes it depends on
- Example:
`javac TicketMachine.java`

Execution

- `C:\bluej\zuul> java Game`
- “java” starts the Java virtual machine.
- The named class is loaded and execution is started.
- Other classes are loaded as needed.
- Only possible if class has been compiled.

Problem: Execute what?

- If we try:

```
C:> java TicketMachine  
Exception in thread "main"  
java.lang.NoSuchMethodError: main
```
- The problem: how does the system know which method to execute?

The main method

- The answer: The java system always executes a method called main with a certain signature:

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

- For this to work, such a method **must exist!**

The main method (2)

- “main” must exist
- “main” must be public
- “main” must be static (class method)
- “main” must have a String array parameter
- Only “main” can be invoked

Main method - example

```
public static void main(String[] args)
{
    TicketMachine tm =
        new TicketMachine(10);
    tm.insertMoney(15);
    tm.printTicket();
}
```

- The main method should
 - create an object
 - call the first method

Command-line arguments

- A java application can accept an arbitrary number of arguments from the command line
 - `java MainClass arg0 arg1 ...`
- Command-line arguments are passed to the main method in an array of strings

Testing

- To test, test drivers must be written.
- All test method calls must be written into a test method.
- All relevant parameter combination must be tested.
- If tests depend on the results of earlier tests, the test driver must be edited and recompiled.
- The test driver must create the objects.

Acknowledgement

The original authors of these slides are the authors of the textbook. The instructor made necessary modifications, with permissions from the authors.