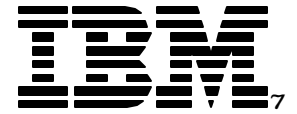


Session:



Lab: Introduction to Java

IBM  server iSeries 400 and AS/400e

Susan Funk
IBM Rochester

© Copyright IBM Corporation, 2001. All Rights Reserved.

This publication may refer to products that are not currently available in your country. IBM makes no commitment to make available any products referred to herein.

IBM  server. For the next generation of e-business.

Lab: Introduction to Java

IBM  server iSeries

8 Copyright International Business Machines Corporation 2001

References in this document to IBM products or services do not imply that IBM intends to make them available in every country.

The following terms are trademarks or registered trademarks of International Business Machines Corporation in the United States, other countries, or both:

AIX	IBM
AS/400	IBM Logo
AS/400e	iSeries
DB2	OS/400
e-business logo	VisualAge

Java and all Java-based trademarks and logos are trademarks or registered trademarks of Sun Microsystems, Inc. in the United States, other countries, or both.

Microsoft, Windows, Windows NT, and the Windows logo are trademarks of Microsoft Corporation in the United States, other countries, or both.

Other company, product and service names may be trademarks or service marks of others.

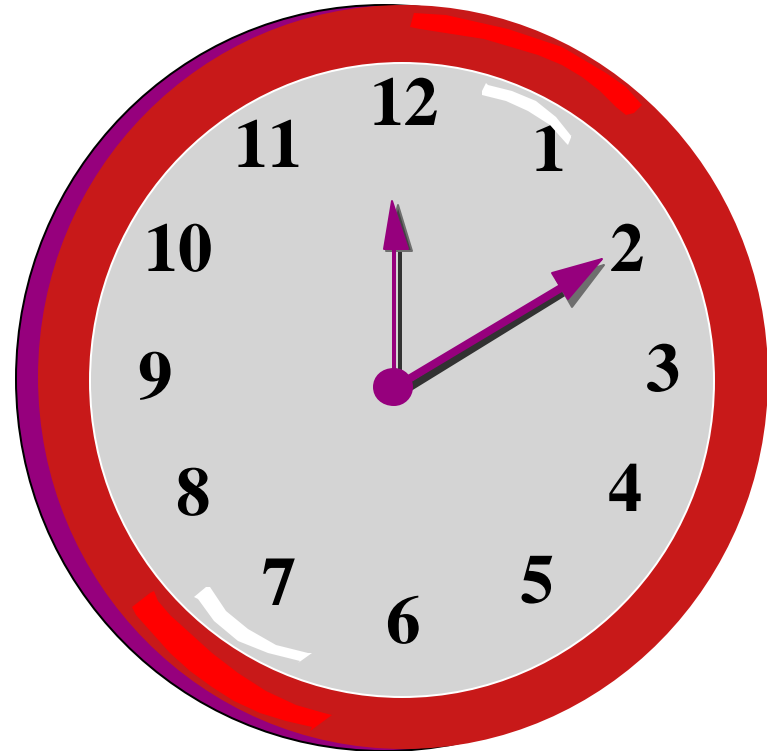
Information is provided "AS IS" without warranty of any kind.

IBM  server. For the next generation of e-business.

Agenda

IBM  server iSeries

- Introduction
- Buzzwords
- Environments
- Program structure
- Syntax
- Development tools
- References
- Lab exercises

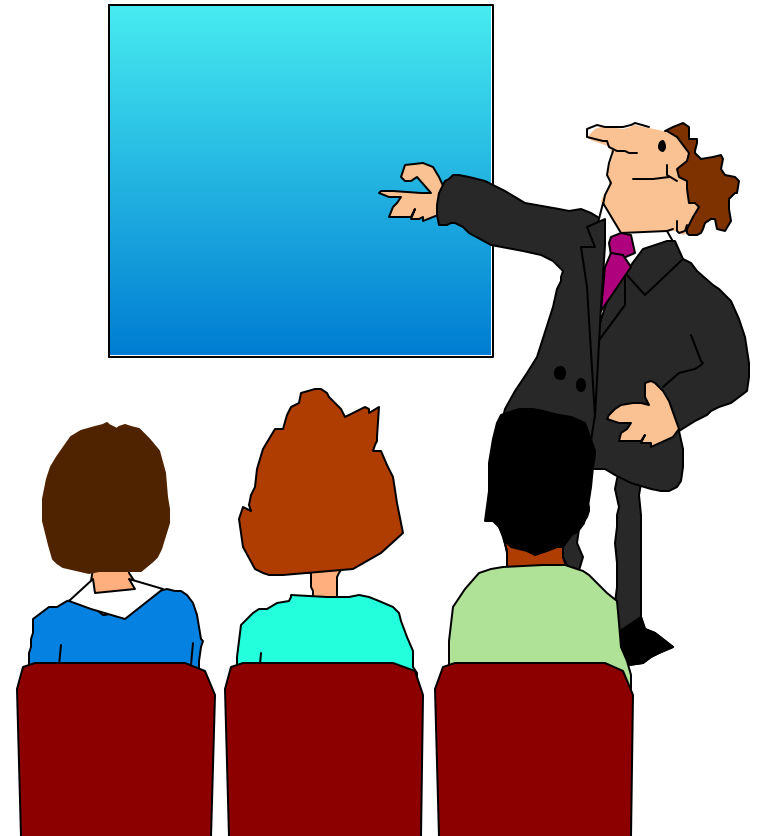


IBM  server. For the next generation of e-business.

Buzzwords

IBM  server iSeries

- *"Simple"*
 - Syntax is based on C++
 - Avoids complicated aspects of C++:
 - ▶ no pointer arithmetic, no memory management
 - Many reusable objects available
- *Object oriented*
 - Classes, objects, methods, inheritance
- *Distributed*
 - TCP/IP, HTTP communication built in
 - Applet and servlet programming models
 - "Internet-ready"
 - ▶ (it is great for non-Internet apps, too!)

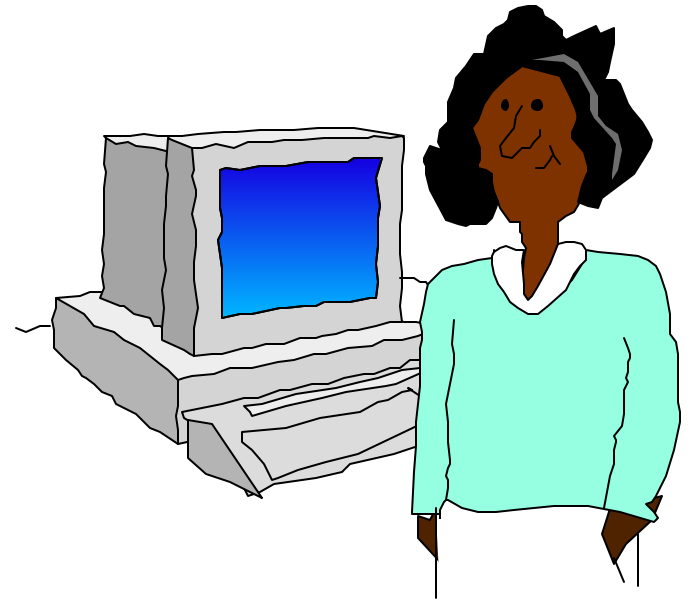


IBM  server. For the next generation of e-business.

Buzzwords

IBM  server iSeries

- *Robust*
 - Array bounds checking
 - Automatic garbage collection
- *Secure*
 - Applet sandbox
 - Class verifier - prevents "forged" Java code
- *Other built in features*
 - Threads
 - Graphical user interface components
 - Collections
 - Input/output
 - Exception handling
 - Internationalization (national language support)



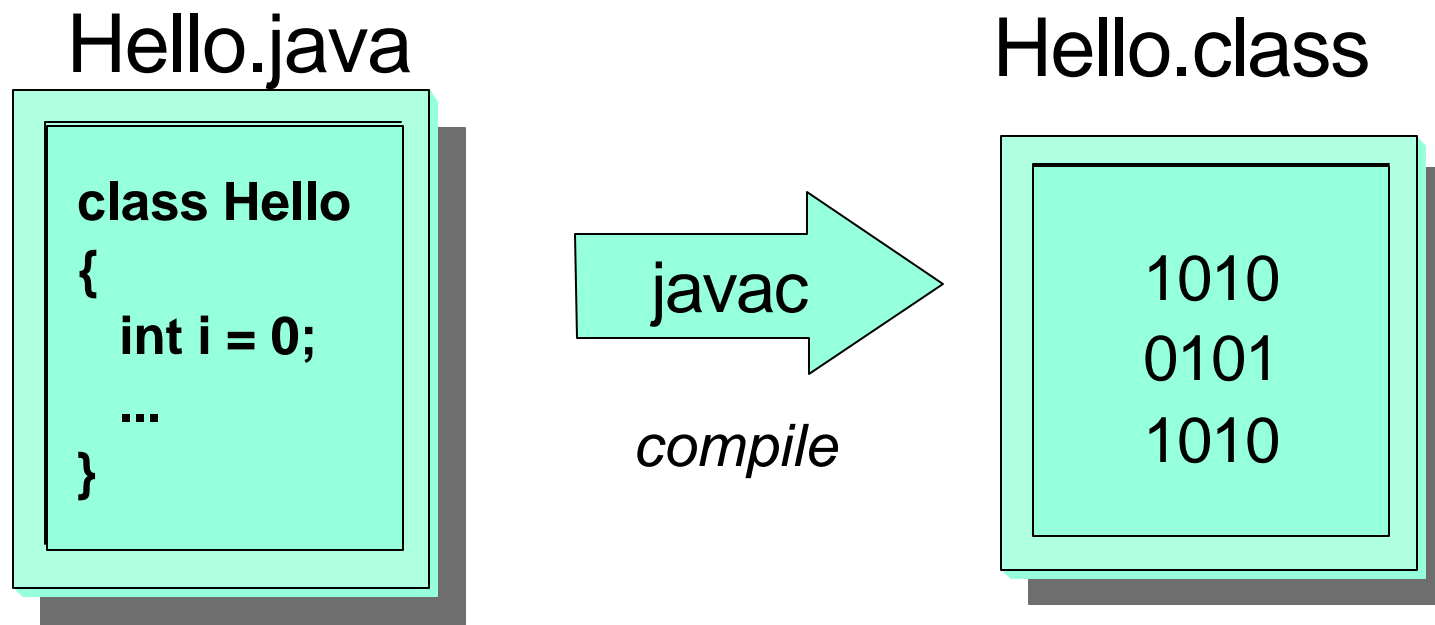
IBM  server. For the next generation of e-business.

Buzzwords

IBM @server iSeries

- *Portable*

- Source code (.java file) edited and compiled on any platform
- Compiled bytecodes (.class file) run on any platform
- All aspects of pure Java are portable (GUI, threads, networking, I/O, etc.)

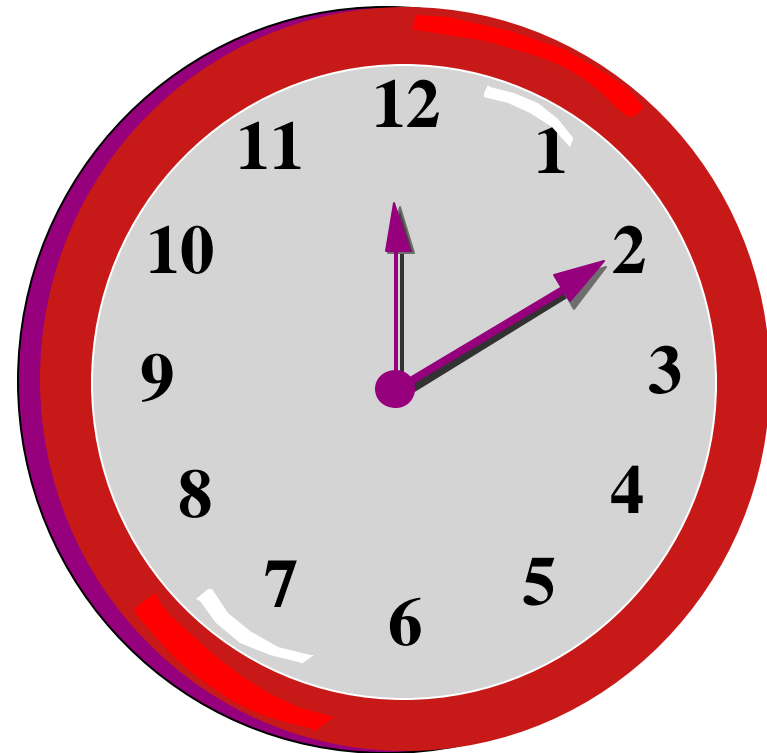


IBM @server. For the next generation of e-business.

Agenda

IBM @server iSeries

- Introduction
- Buzzwords
- Environments
- Program structure
- Syntax
- Development tools
- References
- Lab exercises

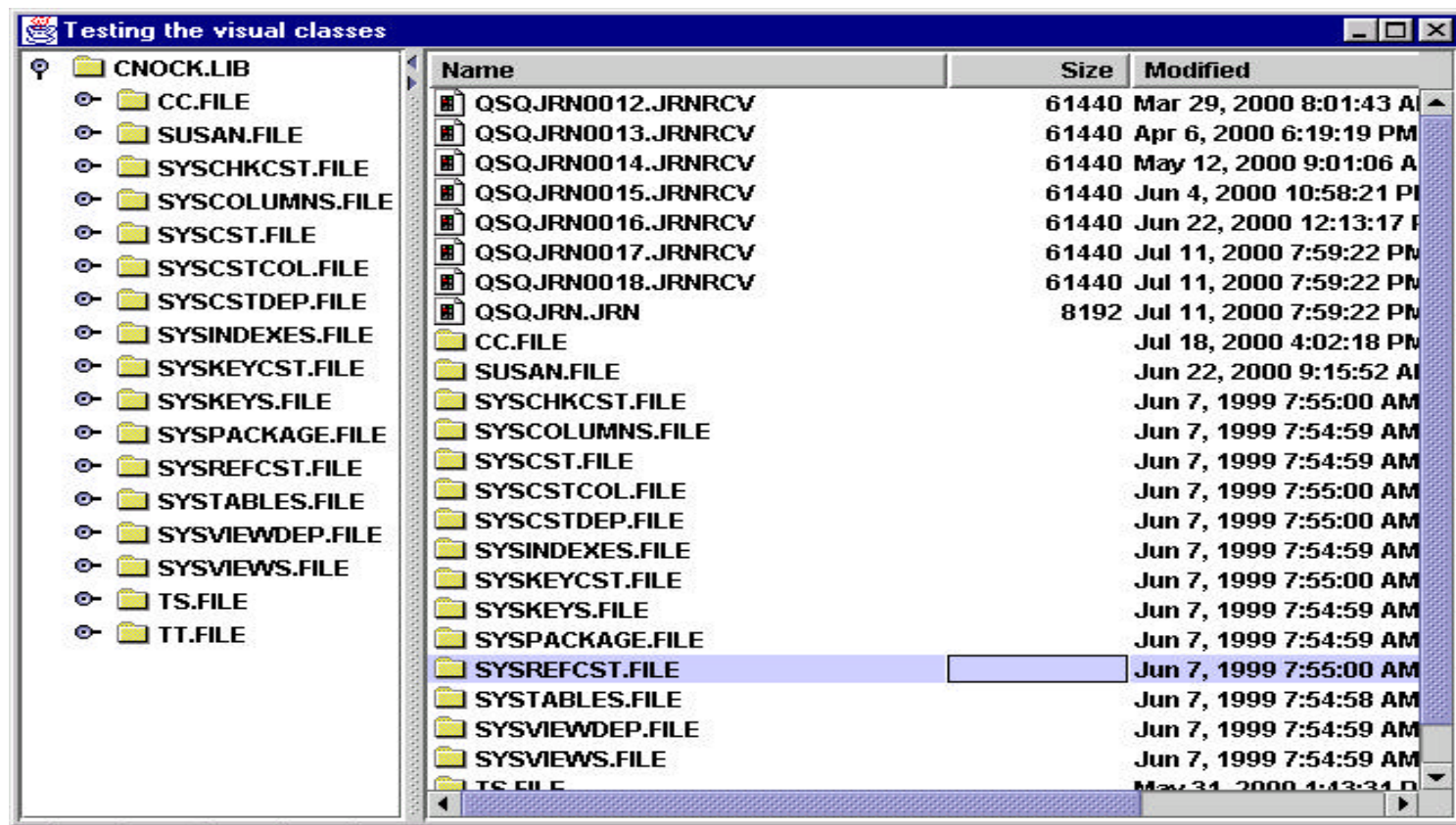


IBM @server. For the next generation of e-business.

Environments: Applications

IBM @server iSeries

- Program is loaded from local or network storage
- Runs on local processor
- Can do just about anything!

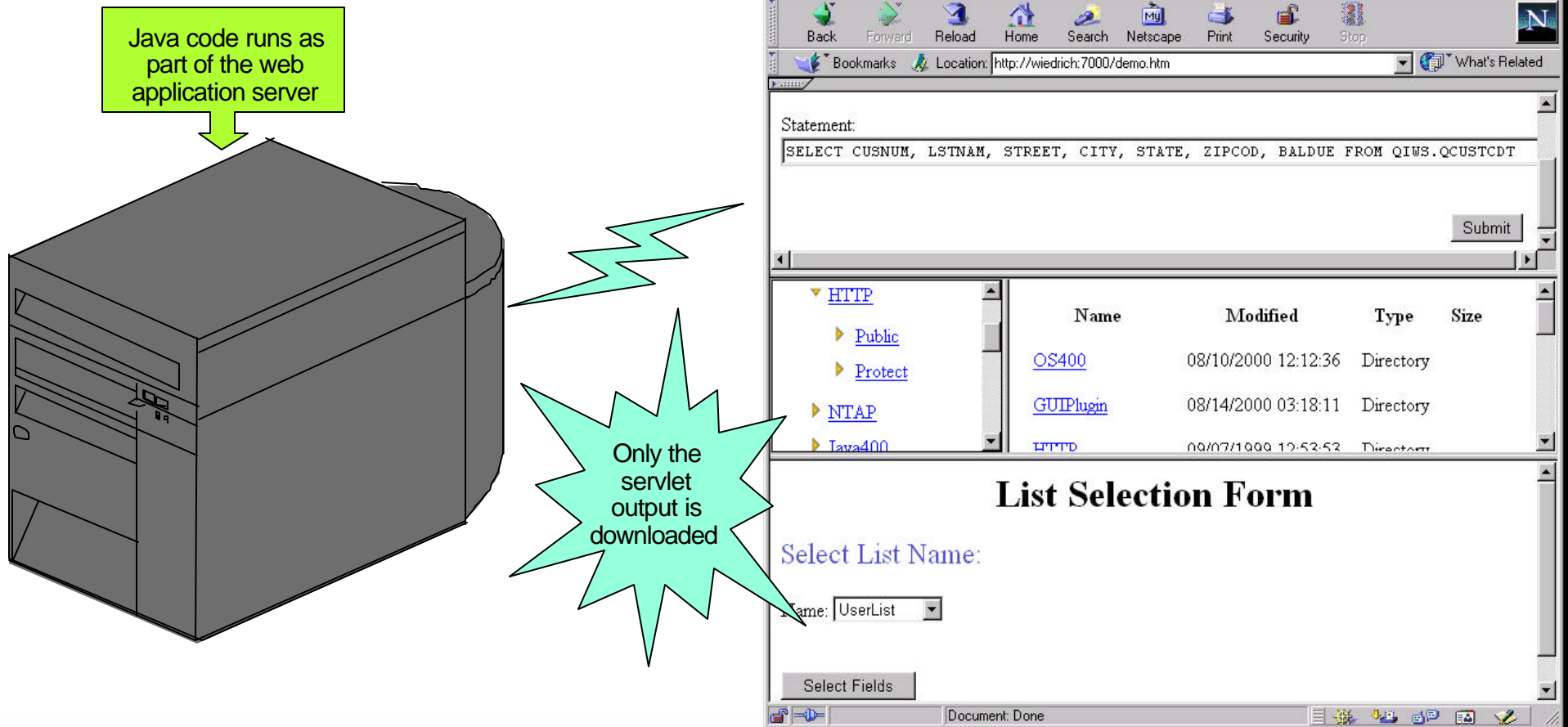


IBM @server. For the next generation of e-business.

Environments: Servlets

IBM @server iSeries

- Runs on web server in response to an HTTP request (e.g. <http://www.mysite.com/info>)
- Output is HTML (not GUI) - the HTML is sent back to the client and presented in the browser



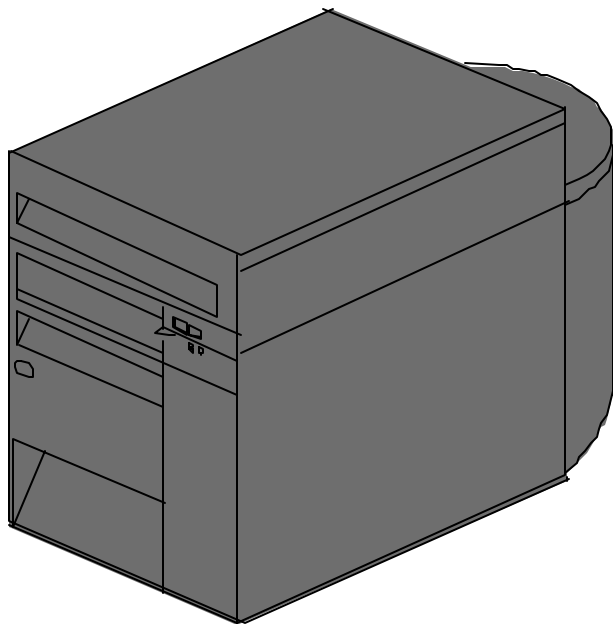
IBM @server. For the next generation of e-business.

Environments: Applets

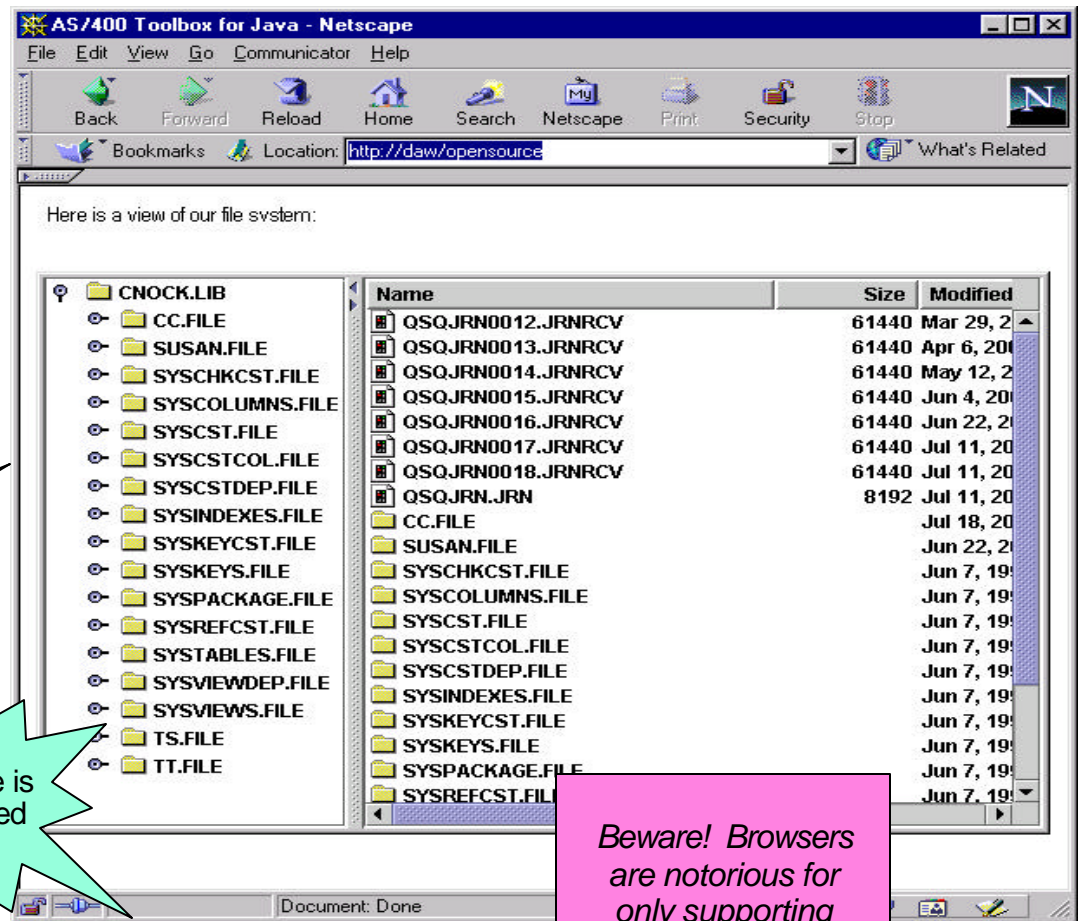
IBM  server iSeries

- Program is downloaded as part of a web page, along with text and images
- Runs on local processor, but always within the context of the web browser
- Input and output via GUI
- Extra security measures:
 - limited file access
 - limited network access

Java code runs as part of the browser



Java code is downloaded



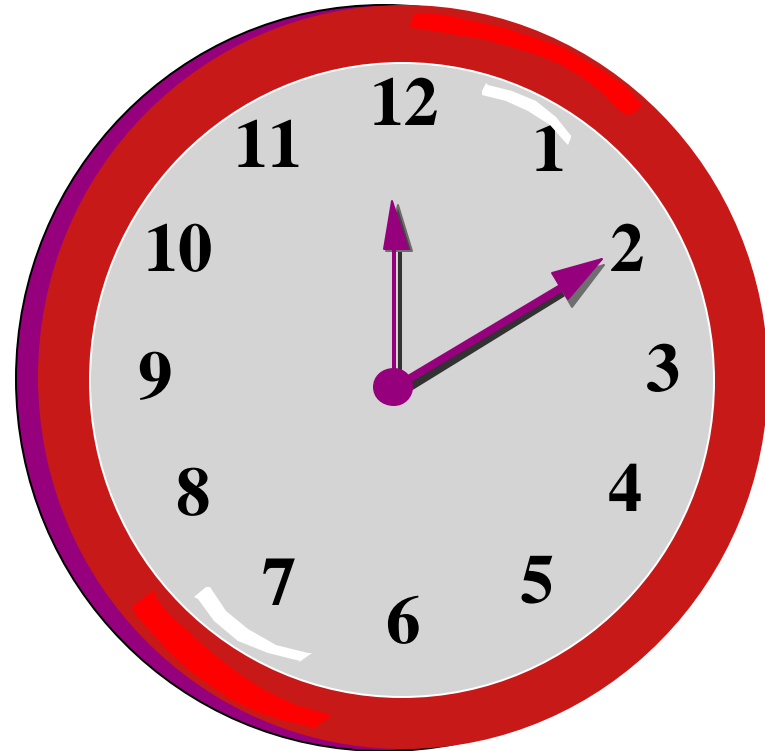
Beware! Browsers are notorious for only supporting backlevel versions of Java

Java != JavaScript
IBM  server For the next generation of e-business.

Agenda

IBM  server iSeries

- Introduction
- Buzzwords
- Environments
- Program structure
- Syntax
- Development tools
- References
- Lab exercises



IBM  server. For the next generation of e-business.

Program structure

ReportGenerator.java

IBM @server iSeries

Imports instruct the compiler where to find other classes that are used.

A **class** defines the variables and methods for a type of **object**.

A **constructor** is a special method used for initializing an object. Its name matches the class name.

All data and methods are strongly-typed. Java compilers are very strict!

```
import java.util.Date;

public class ReportGenerator
{
    private Date reportDate;
    private String title;

    public ReportGenerator(String theTitle)
    {
        title = theTitle;
        reportDate = new Date(); // The current date and time.
    }

    public void print()
    {
        System.out.println("Report: " + title);
        System.out.println("Generated on: " + reportDate);
        // ... more Java code here ...
    }

    public static void main(String[] args)
    {
        ReportGenerator rg = new ReportGenerator("Marketing expenses");
        rg.print();
    }
}
```

There is usually one **class** per source file. Its name matches the source file name.

A **field** defines the part of the content (data) of the object. A **private field** is usable only within this class. This is called **encapsulation**.

Comments are denoted by // or /* */.

A **method** defines a behavior of the object. It is analogous to a function, procedure, subroutine, etc.

The **main()** method is special - it is the **entry point** for a Java application

IBM @server. For the next generation of e-business.

Syntax

IBM  server iSeries

- Everything in Java is case-sensitive
- Java syntax is based on C++
- Whitespace is not significant
 - Use indentation and whitespace to make code readable
- Everything in Java is case-sensitive
- Variable names can be any length
 - Using descriptive names can lead to self-documenting code
 - Can be any Unicode characters (except for a few)
- Java handles all Strings as Unicode
- *Everything in Java is case-sensitive!*

```
public class HelloWorld
{
    public static void main(String[] args)
    {
        System.out.println("Hello World");
    }
}
```

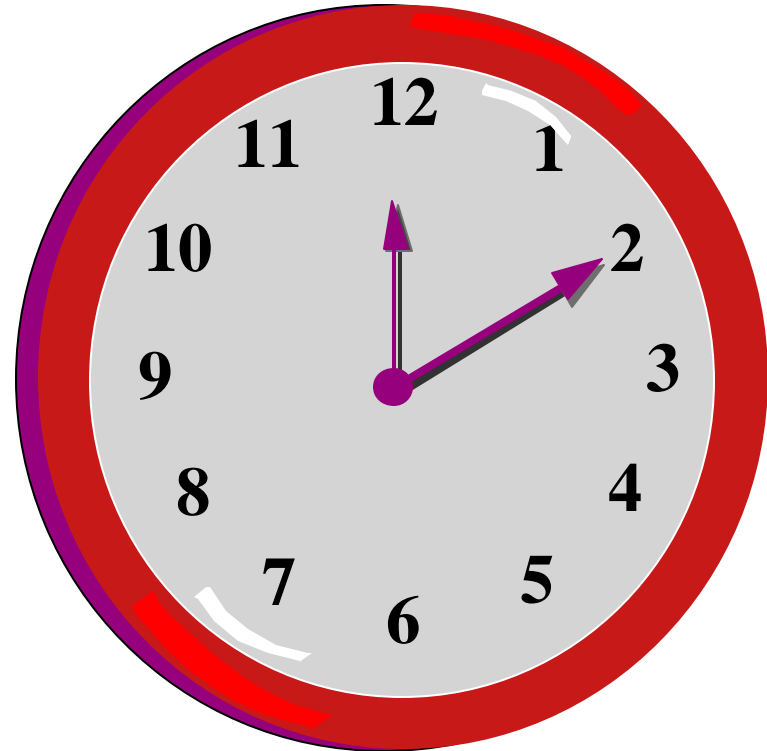
```
public class HelloWorld { public static void main(String[] args) { System.out.println("Hello World"); } }
```

IBM  server. For the next generation of e-business.

Agenda

IBM  server iSeries

- Introduction
- Buzzwords
- Environments
- Program structure
- Syntax
- Development tools
- References
- Lab exercises

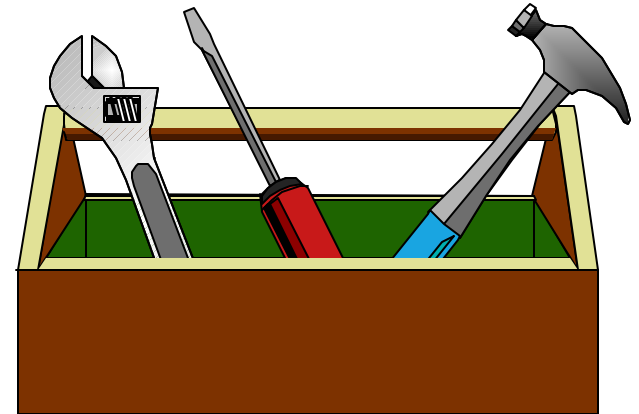


IBM  server. For the next generation of e-business.

Development tools

IBM @server iSeries

- **Java Developers Kit (JDK or SDK)** - includes compiler, JVM, debugger, javadoc, appletviewer, etc.
 - Available from Sun, IBM, others...
 - **The minimum you need for Java development**
- **Compiler** - compiles source code (.java file) into Java bytecodes (.class file)
 - Sun: javac
 - IBM: jikes - much faster!
- **Jar** - Java archive file
 - Used for packaging multiple .class files
- **Java Virtual Machine (JVM)** - runs Java bytecodes
 - Interprets bytecodes and runs as machine instructions
 - Just-In-Time (JIT) compiler - caches machine instructions
 - iSeries stores optimized machine instructions with .class file
 - **The minimum you need to run Java programs**

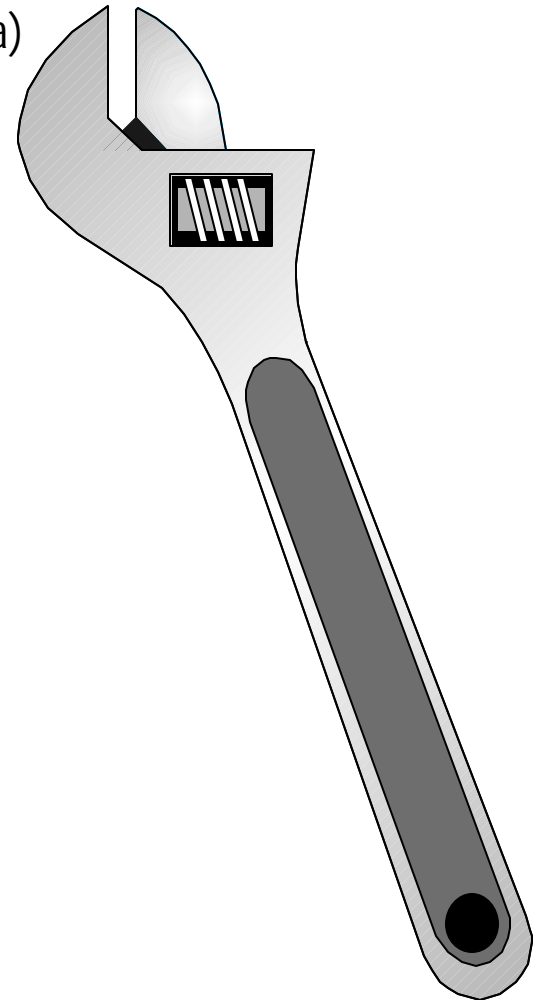


IBM @server. For the next generation of e-business.

Development tools

IBM  server iSeries

- Integrated development environment (IDE)
 - Many available from many vendors (e.g. IBM VisualAge for Java)
 - Includes graphical debugger, visual development
- Toolbox for Java/JTOpen
 - Building blocks specifically for iSeries Java programs
 - Runs on client or server
 - Open source
 - Example code is available



IBM  server. For the next generation of e-business.

References

IBM  server iSeries

- java.sun.com
- ibm.com/java
- ibm.com/iSeries/java
- ibm.com/iSeries/toolbox

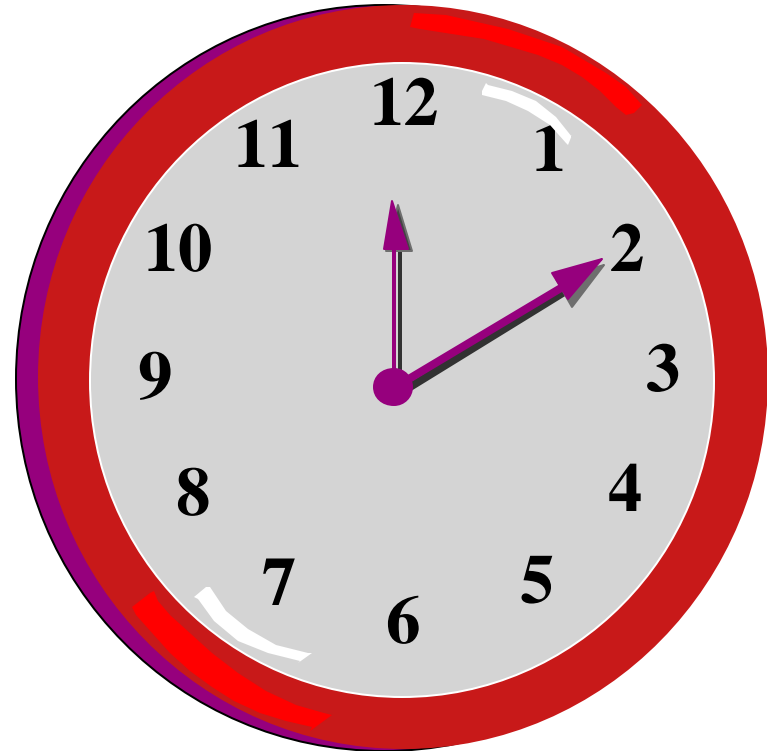


IBM  server. For the next generation of e-business.

Agenda

IBM  server iSeries

- Introduction
- Buzzwords
- Environments
- Program structure
- Syntax
- Development tools
- References
- Lab exercises



IBM  server. For the next generation of e-business.

Questions?

IBM server. For the next generation of e-business.

Lab Exercises

IBM @server. For the next generation of e-business.

Common debugging issues

IBM @server iSeries

Class not found when compiling:

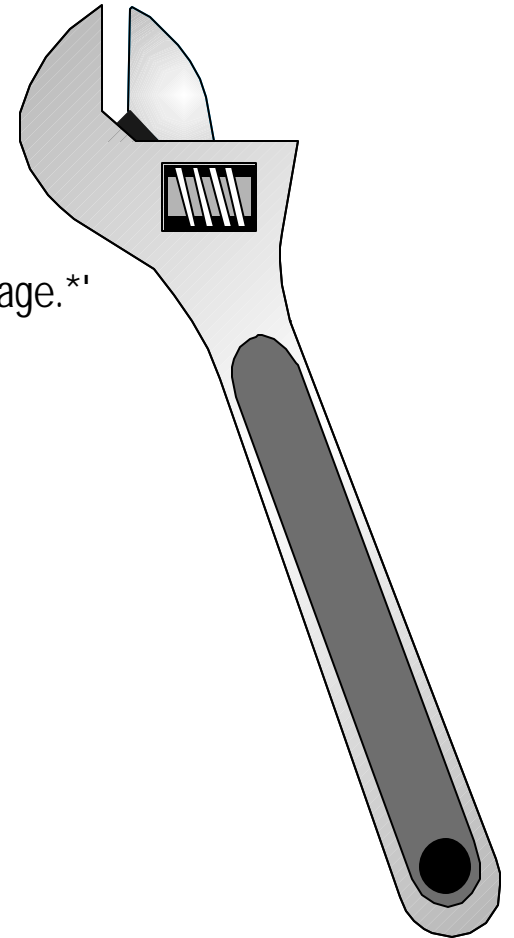
- Either you have not imported this class, or you have not specified it correctly
 - Import this class or import its package with a *
 - For instance, 'import samplePackage.sampleClass or 'import samplePackage.*'
 - Check the exact spelling and capitalization

Not able to read a class when compiling:

- Check the spelling of the java file name
- Make sure you are compiling in the directory where the file resides

Class definition not found when compiling:

- Make sure this class is in a directory in your CLASSPATH (/myPath/)
- If the class is in a jar, include the jar in your CLASSPATH (/myPath/myJar.jar)

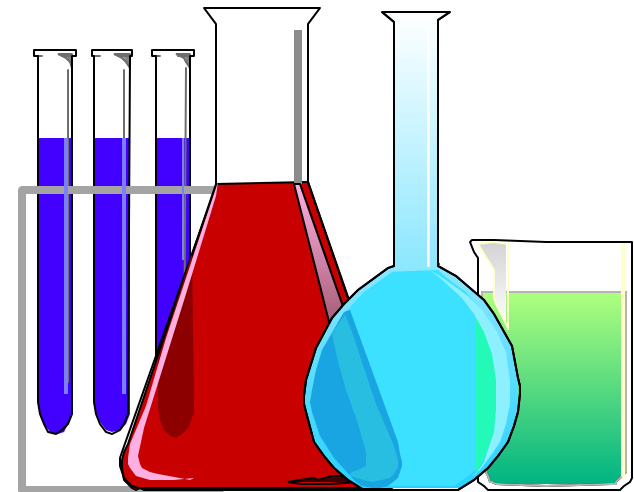


IBM @server. For the next generation of e-business.

Lab exercises

IBM @server iSeries

1. Create and run a simple Java application on Windows
2. Run a simple Java application on an iSeries
3. Optimize an iSeries Java application
4. Create a simple web page
5. Create and run a simple Java applet
6. Use Qshell on an iSeries
7. Create and run a simple Java servlet



Please ask questions!

IBM @server. For the next generation of e-business.