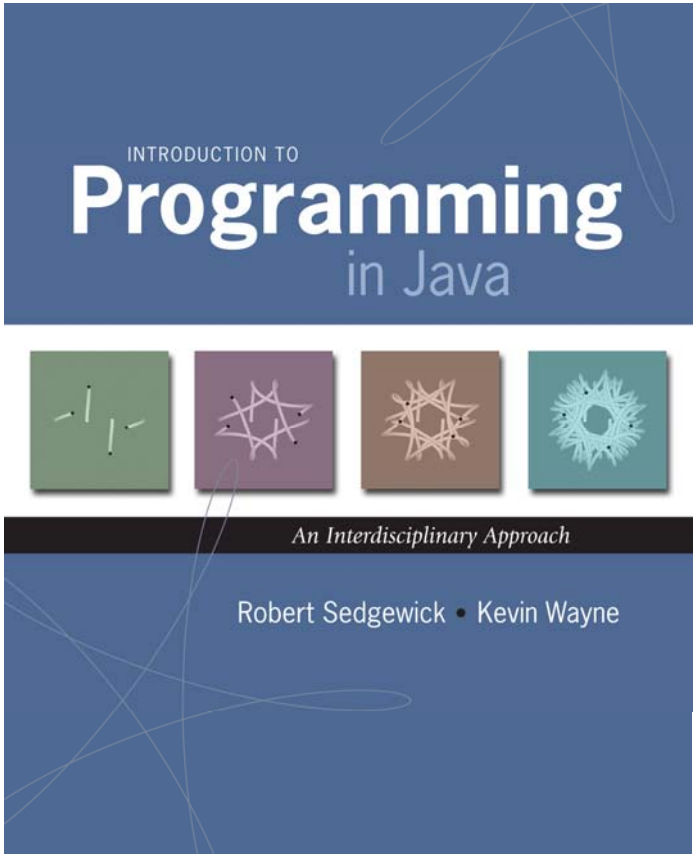
1.1 Your First Program



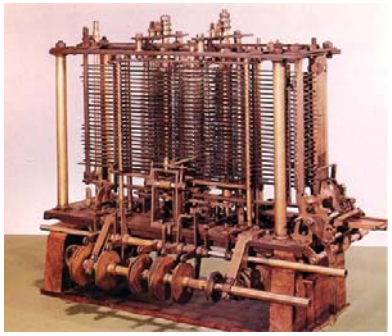
Why Programming?

Idealized computer. "Please simulate the motion of a system of N heavenly bodies, subject to Newton's laws of motion and gravity."

Prepackaged software solutions. Great, if it does exactly what you need.

Computer programming. Art of making a computer do what you want.

Ada Lovelace



Analytic Engine

Languages

> *"Instead of imagining that our main task is to instruct a computer what to do, let us concentrate rather on explaining to human beings what we want a computer to do."* – *Donald Knuth*

Machine languages. Tedious and error-prone.

Natural languages. Ambiguous and hard for computer to parse.

Kids Make Nutritious Snacks.
Red Tape Holds Up New Bridge.
Police Squad Helps Dog Bite Victim.
Local High School Dropouts Cut in Half.

[real newspaper headlines, compiled by Rich Pattis]

High-level programming languages. Acceptable tradeoff.

Why Java?

Java features.

- Widely used.
- Widely available.
- Embraces full set of modern abstractions.
- Variety of automatic checks for mistakes in programs.

Java economy.

- Mars rover.
- Cell phones.
- Blu-ray Disc.
- Web servers.
- Medical devices.
- Supercomputing.
- ...

$100 billion,
5 million developers

James Gosling
http://java.net/jag

Why Java?

Java features.

- Widely used.
- Widely available.
- Embraces full set of modern abstractions.
- Variety of automatic checks for mistakes in programs.

Caveat.

> *"There are only two kinds of programming languages: those people always [gripe] about and those nobody uses."* – *Bjarne Stroustrup*

Why Java?

Java features.

- Widely used.
- Widely available.
- Embraces full set of modern abstractions.
- Variety of automatic checks for mistakes in programs.

Caveat. No perfect language.

Our approach.

- Minimal subset of Java.
- Develop general programming skills that are applicable to:
  C, C++, C#, Perl, Python, Ruby, Matlab, Fortran, Fortress, ...

A Rich Subset of the Java Language

Built-In Types	
int	double
long	String
char	boolean

System
System.out.println()
System.out.print()
System.out.printf()

Math Library	
Math.sin()	Math.cos()
Math.log()	Math.exp()
Math.sqrt()	Math.pow()
Math.min()	Math.max()
Math.abs()	Math.PI

Flow Control	
if	else
for	while

Parsing
Integer.parseInt()
Double.parseDouble()

Primitive Numeric Types		
+	-	*
/	%	++
--	>	<
<=	>=	==
!=		

Boolean	
true	false
||	&&
!	

Punctuation	
{	}
(	)
,	;

Assignment
=

String	
+	""
length()	compareTo()
charAt()	matches()

Arrays
a[i]
new
a.length

Objects	
class	static
public	private
toString()	equals()
new	main()

Create, Compile, Execute

Programming in Java.

- Create the program by typing it into a text editor, and save it as `HelloWorld.java`

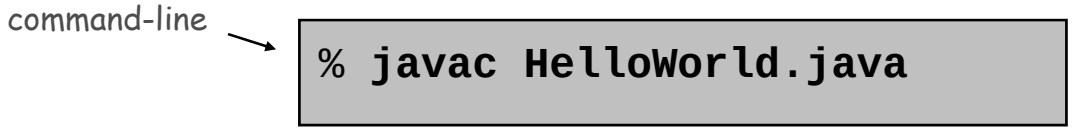
```
/*************************************************
 * Prints "Hello, World"
 * Everyone's first Java program.
 *************************************************/

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

HelloWorld.java

Programming in Java.

- Create the program by typing it into a text editor, and save it as `HelloWorld.java`
- Compile it by typing at the command-line:
  `javac HelloWorld.java`



(or click the Compile button in DrJava)

- This creates a Java bytecode file named: `HelloWorld.class`

Programming in Java.

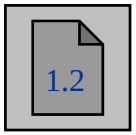
- Create the program by typing it into a text editor, and save it as `HelloWorld.java`
- Compile it by typing at the command-line:
  `javac HelloWorld.java`
- Execute it by typing at the command-line:
  `java HelloWorld`

command-line →

```
% javac HelloWorld.java

% java HelloWorld
Hello, World
```

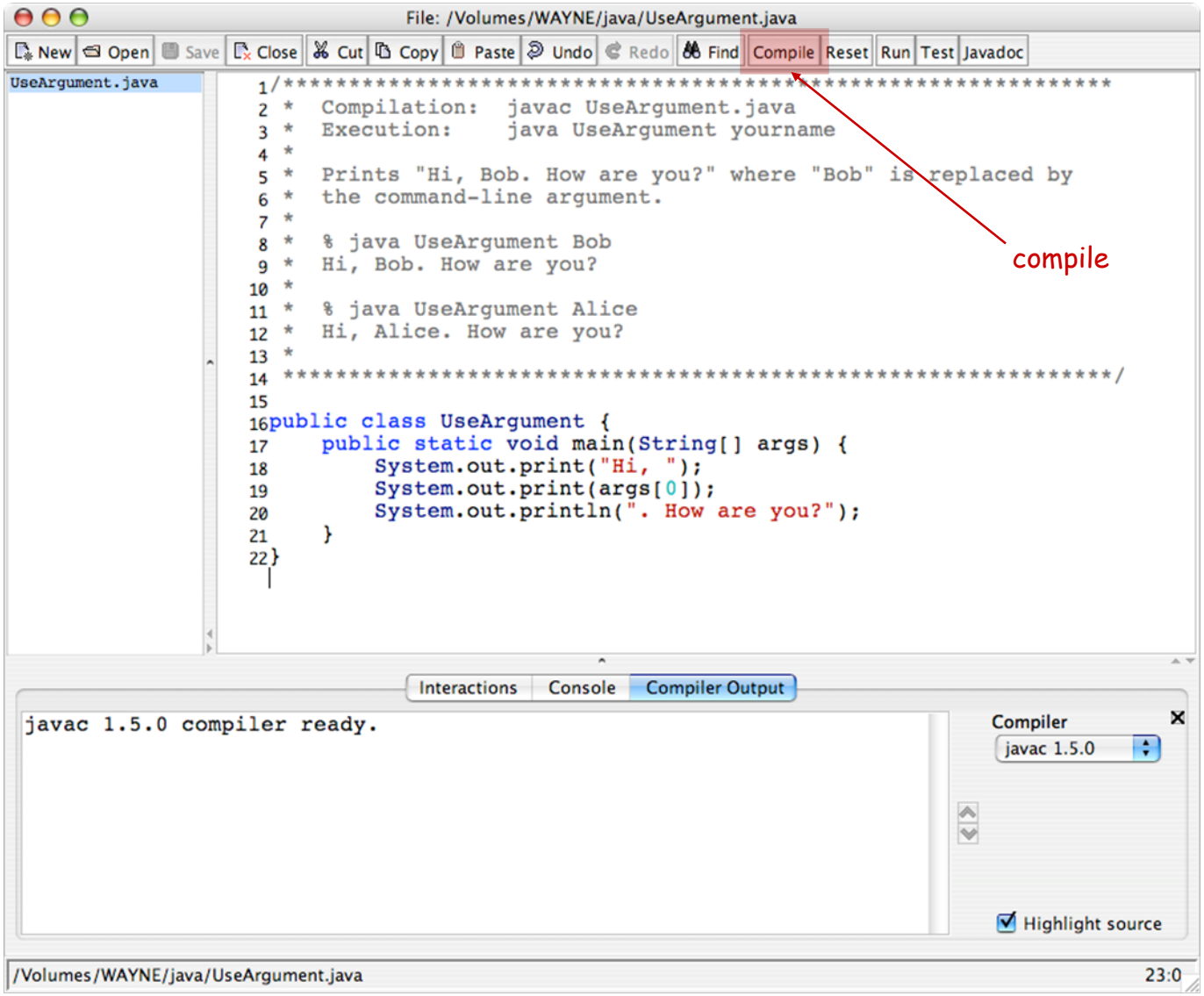
(or click the Run button in DrJava)



Dr. Java

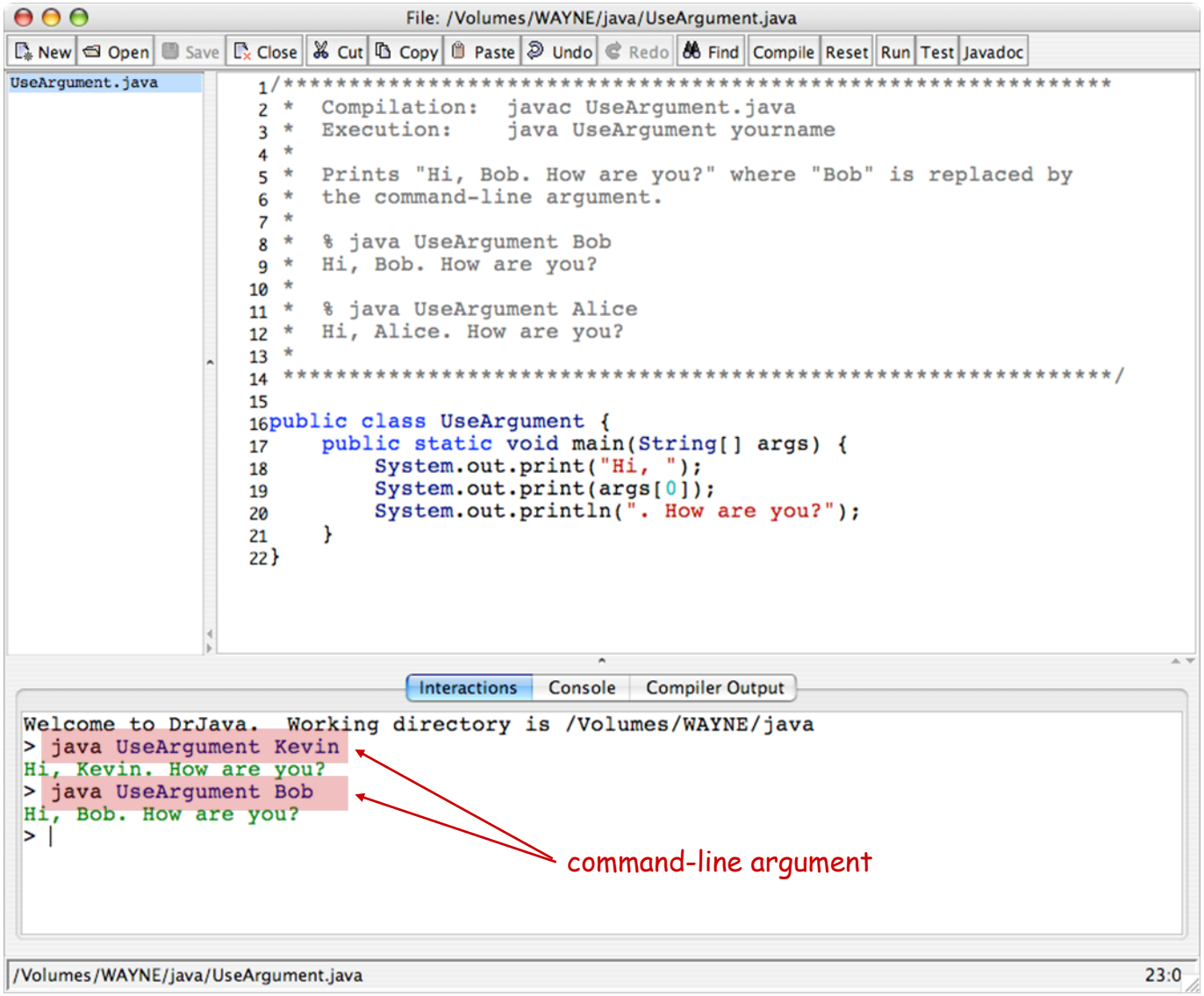
http://drjava.org

Dr. Java




```
1 /*************************************************************************
2  *  Compilation:  javac UseArgument.java
3  *  Execution:    java UseArgument yourname
4  *
5  *  Prints "Hi, Bob. How are you?" where "Bob" is replaced by
6  *  the command-line argument.
7  *
8  *  % java UseArgument Bob
9  *  Hi, Bob. How are you?
10 *
11 *  % java UseArgument Alice
12 *  Hi, Alice. How are you?
13 *
14 *************************************************************************/
15
16public class UseArgument {
17    public static void main(String[] args) {
18        System.out.print("Hi, ");
19        System.out.print(args[0]);
20        System.out.println(". How are you?");
21    }
22}
```

Dr. Java


File: /Volumes/WAYNE/java/UseArgument.java
New
Open
Save
Close
Cut
Copy
Paste
Undo
Redo
Find
Compile
Reset
Run
Test
Javadoc
UseArgument.java
1/***
2 * Compilation: javac UseArgument.java
3 * Execution: java UseArgument yourname
4 *
5 * Prints "Hi, Bob. How are you?" where "Bob" is replaced by
6 * the command-line argument.
7 *
8 * % java UseArgument Bob
9 * Hi, Bob. How are you?
10 *
11 * % java UseArgument Alice
12 * Hi, Alice. How are you?
13 *
14 ***/
15
16public class UseArgument {
17 public static void main(String[] args) {
18 System.out.print("Hi, ");
19 System.out.print(args[0]);
20 System.out.println(". How are you?");
21 }
22}
Interactions
Console
Compiler Output
Welcome to DrJava. Working directory is /Volumes/WAYNE/java
> java UseArgument Kevin
Hi, Kevin. How are you?
> java UseArgument Bob
Hi, Bob. How are you?
>
command-line argument
/Volumes/WAYNE/java/UseArgument.java
23:0

Java Features

Java is:

- Object oriented.
- Statically typed.
- Architecture neutral.
- Multi-threaded.
- Garbage-collecting.
- Robust.
- Small.
- Simple.
- Fast.
- Secure.
- Extensible.
- Well-understood.
- Fun.

Java is:

- Not new.
- Designed for toasters.
- Not done yet.
- Not as useful as C, C++, FORTRAN.
- Slow.
- Unsafe.
- Huge.
- Complex.

Don't believe anything on this slide! Make up your own mind.

Java Bytecode

```
0000000 312 376 272 276  \0  \0  \0   .  \0 035  \n  \0 006  \0 017  \t
0000020  \0 020  \0 021  \b  \0 022  \n  \0 023  \0 024 007  \0 025 007
0000040  \0 026 001  \0 006   <   i   n   i   t   > 001  \0 003   (   )
0000060   V 001  \0 004   C   o   d   e 001  \0 017   L   i   n   e   N
0000100   u   m   b   e   r   T   a   b   l   e 001  \0 004   m   a   i
0000120   n 001  \0 026   (   [   L   j   a   v   a   /   l   a   n   g
0000140   /   S   t   r   i   n   g   ;   )   V 001  \0  \n   S   o   u
0000160   r   c   e   F   i   l   e 001  \0 017   H   e   l   l   o   W
0000200   o   r   l   d   .   j   a   v   a  \f  \0 007  \0  \b 007  \0
0000220 027  \f  \0 030  \0 031 001  \0  \f   H   e   l   l   o   ,
0000240   W   o   r   l   d 007  \0 032  \f  \0 033  \0 034 001  \0  \n
0000260   H   e   l   l   o   W   o   r   l   d 001  \0 020   j   a   v
0000300   a   /   l   a   n   g   /   O   b   j   e   c   t 001  \0 020
0000320   j   a   v   a   /   l   a   n   g   /   S   y   s   t   e   m
0000340 001  \0 003   o   u   t 001  \0 025   L   j   a   v   a   /   i
0000360   o   /   P   r   i   n   t   S   t   r   e   a   m   ; 001  \0
0000400 023   j   a   v   a   /   i   o   /   P   r   i   n   t   S   t
0000420   r   e   a   m 001  \0 007   p   r   i   n   t   l   n 001  \0
0000440 025   (   L   j   a   v   a   /   l   a   n   g   /   S   t   r
0000460   i   n   g   ;   )   V  \0   !  \0 005  \0 006  \0  \0  \0  \0
0000500  \0 002  \0 001  \0 007  \0  \b  \0 001  \0  \t  \0  \0  \0 035
0000520  \0 001  \0 001  \0  \0  \0 005   * 267  \0 001 261  \0  \0  \0
0000540 001  \0  \n  \0  \0  \0 006  \0 001  \0  \0  \0  \f  \0  \t  \0
0000560 013  \0  \f  \0 001  \0  \t  \0  \0  \0   %  \0 002  \0 001  \0
0000600  \0  \0  \t 262  \0 002 022 003 266  \0 004 261  \0  \0  \0 001
0000620  \0  \n  \0  \0  \0  \n  \0 002  \0  \0  \0 017  \0  \b  \0 020
0000640  \0 001  \0  \r  \0  \0  \0 002  \0 016
0000652
```

HelloWorld.class