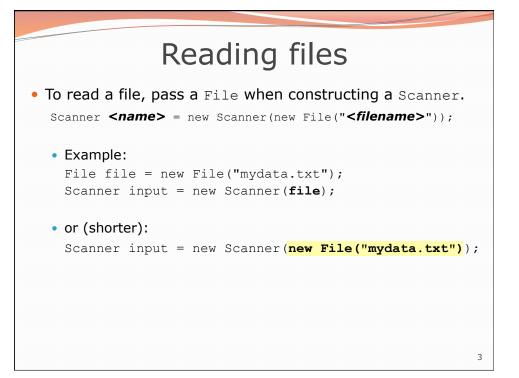
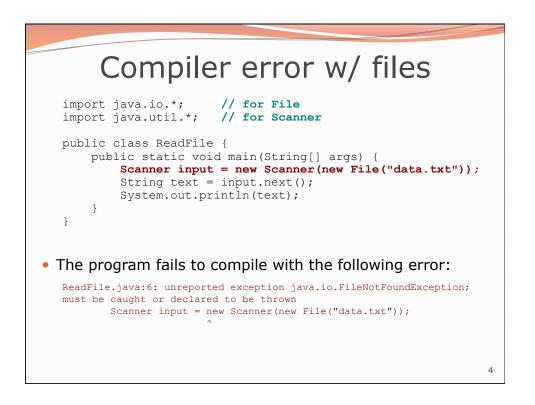
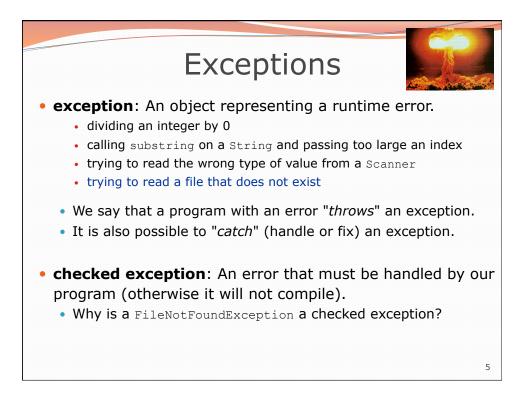
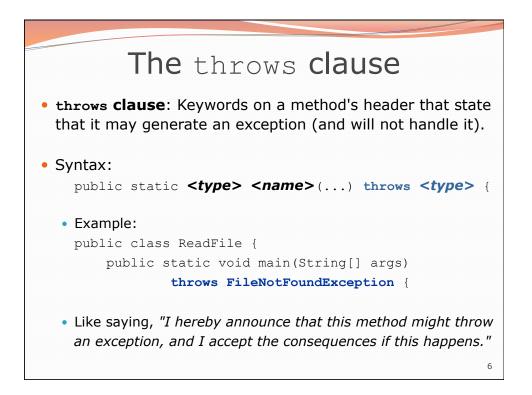


Input/output (I/O)		
<pre>import java.io.*;</pre>		
 (This doesn't actua File f = new Fi 	<pre>ect to get info about a file on your o lly create a new file on the hard disk.) lle("example.txt"); && f.length() > 1000) { </pre>	drive.
Method name	Description	1
canRead()	returns whether file is able to be read	
delete()	removes file from disk	
exists()	whether this file exists on disk	
getName()	returns file's name	
length()	returns number of bytes in file	
-		2

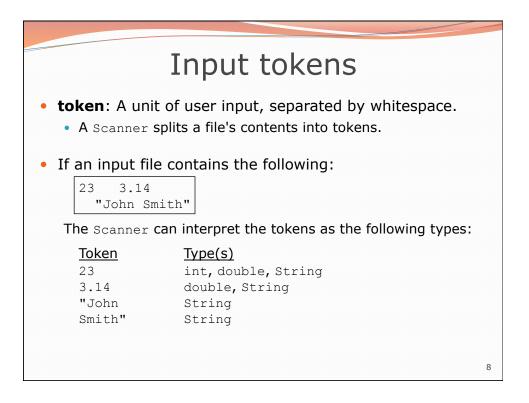


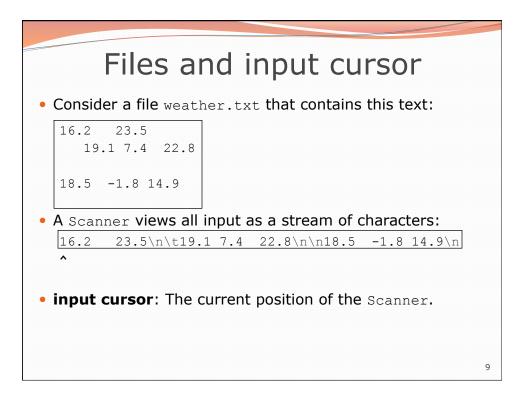


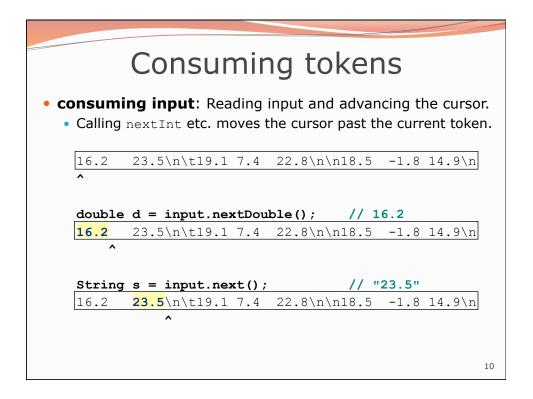


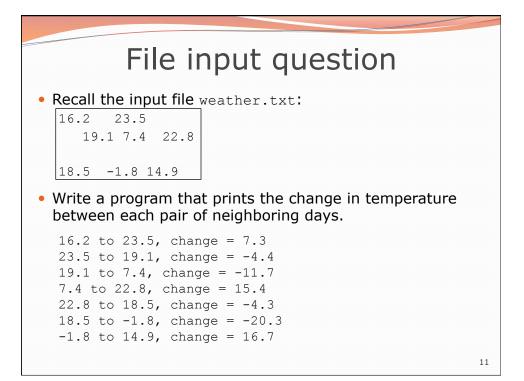


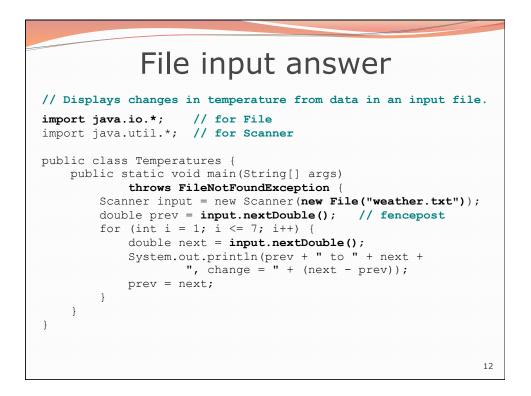


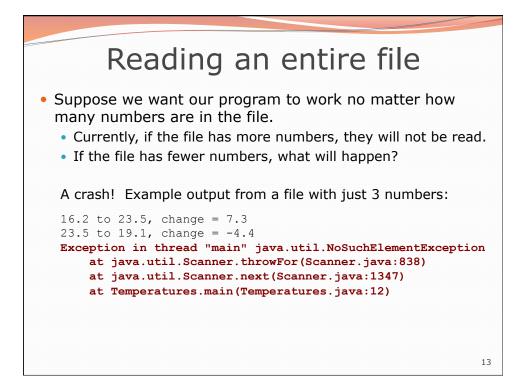


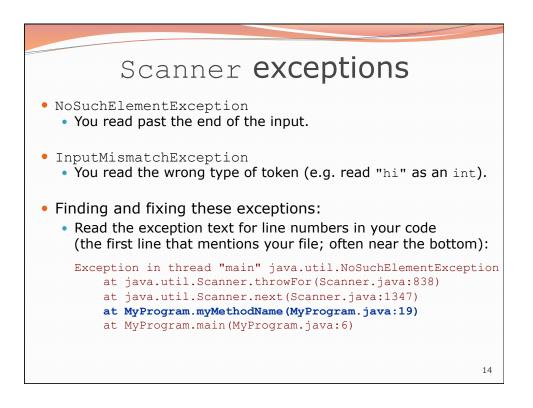












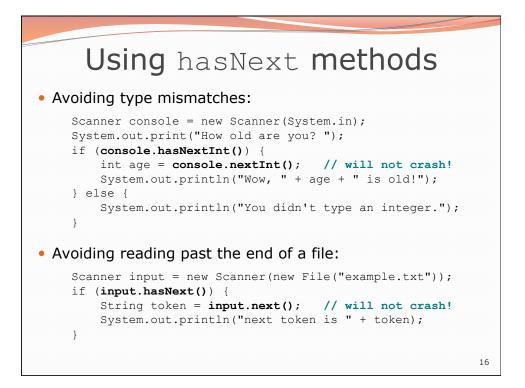
Scanner tests for valid input

Method	Description	
hasNext()	returns true if there is a next token	
hasNextInt()	returns true if there is a next token and it can be read as an int	
hasNextDouble()	returns true if there is a next token and it can be read as a double	

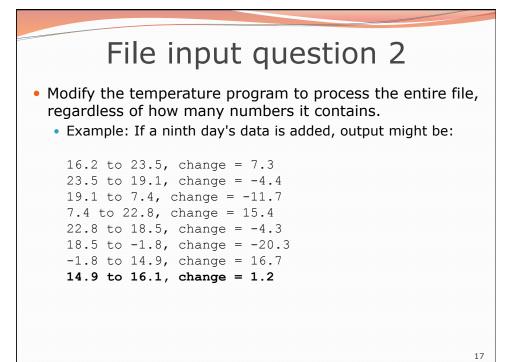
- These methods of the Scanner do not consume input; they just give information about what the next token will be.
 - Useful to see what input is coming, and to avoid crashes.

• These methods can be used with a console Scanner, as well.

• When called on the console, they sometimes pause waiting for input.



15



File input answer 2
<pre>// Displays changes in temperature from data in an input file. import java.io.*; // for File import java.util.*; // for Scanner</pre>
<pre>public class Temperatures { public static void main(String[] args) throws FileNotFoundException { Scanner input = new Scanner(new File("weather.txt")); double prev = input.nextDouble(); // fencepost while (input.hasNextDouble()) { double next = input.nextDouble(); System.out.println(prev + " to " + next +</pre>
18

