# Text File Input and Output

# Objectives

You will be able to

- Write text files from your Java programs.
- Read text files in your Java programs.

# Reading a Text File

- Reading a text file is similar to reading text from the keyboard.
  - Create a Scanner object.
  - Use methods of the Scanner object to read text from the file.

# Reading a Text File

- Differences
  - We have to create a *File* object and pass it to the Scanner constructor.
    - Identifies the file to be read.
  - File input always ends at some point – End of File
    - Keyboard input never ends!
  - We have to check if there is more data before reading.

- More things can go wrong.
  - The file we ask to read may not exist!
  - *Exceptions* may be thrown.

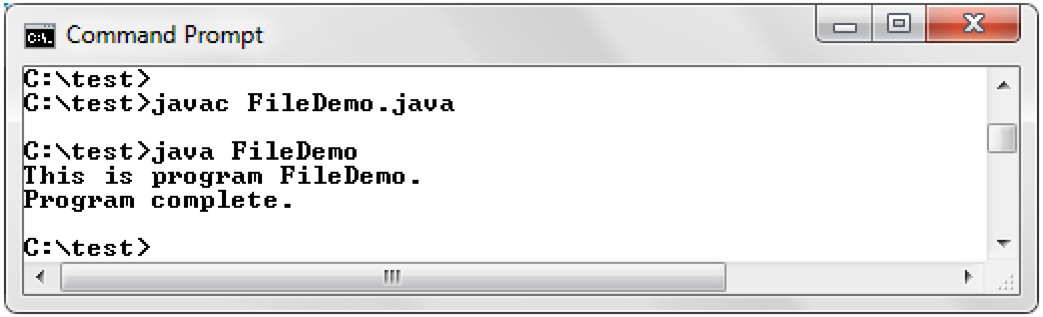
# Program FileDemo

- Demonstrates text file input by reading a file specified by the user and outputting the contents to the screen.

- Start with "Hello, world!"
- Add functionality in tiny steps.

```
/*********************************************************
*
* Program FileDemo.java
*
* Author: Rollins Turner
*
* This program reads a text file and outputs the contents
* to the screen.
*
**********************************************************/

import java.util.Scanner;
import java.io.File;

public class FileDemo
{
    public static void main( String[] args)
    {
        System.out.println("This is program FileDemo.")

        System.out.println("Program complete.")
    }
}
```

# Program Running



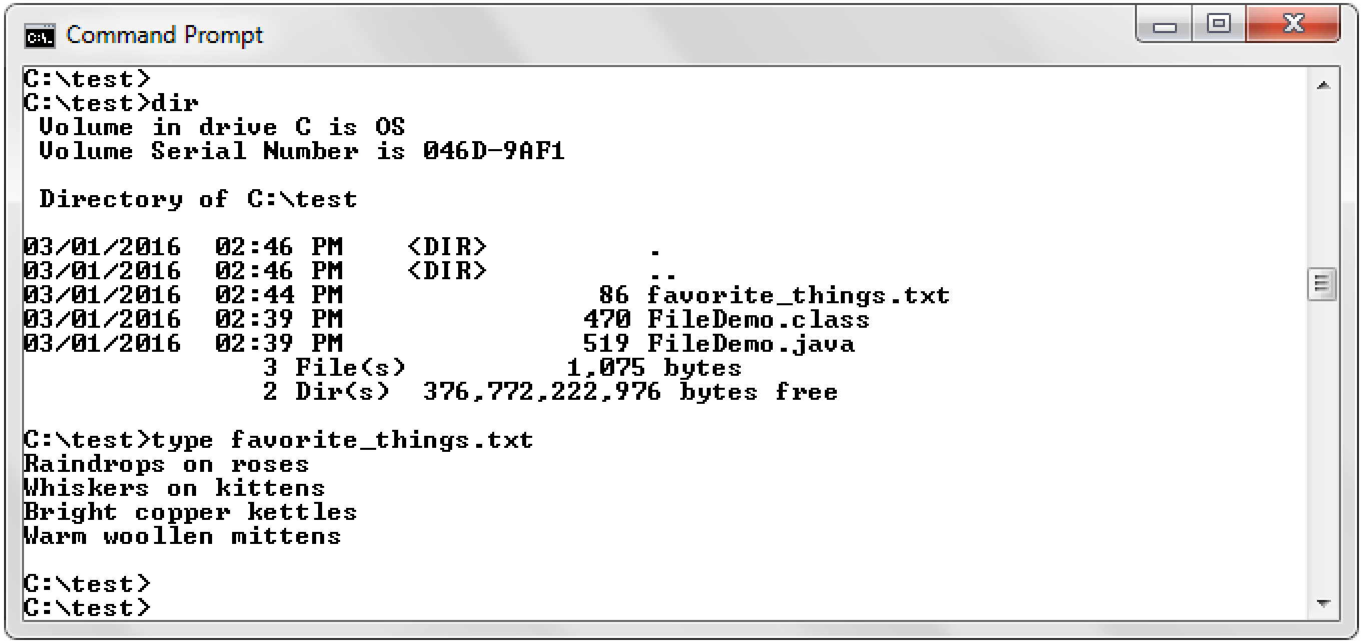
```
C:\test>
C:\test>javac FileDemo.java

C:\test>java FileDemo
This is program FileDemo.
Program complete.

C:\test>
```

# File Demo

- We need a text file to read.

- Download file favorite_things.txt  from the class web site into your test directory.
- http://www.cse.usf.edu/~turnerr/Programming_Concepts/Downloads/

# File favorite_things.txt

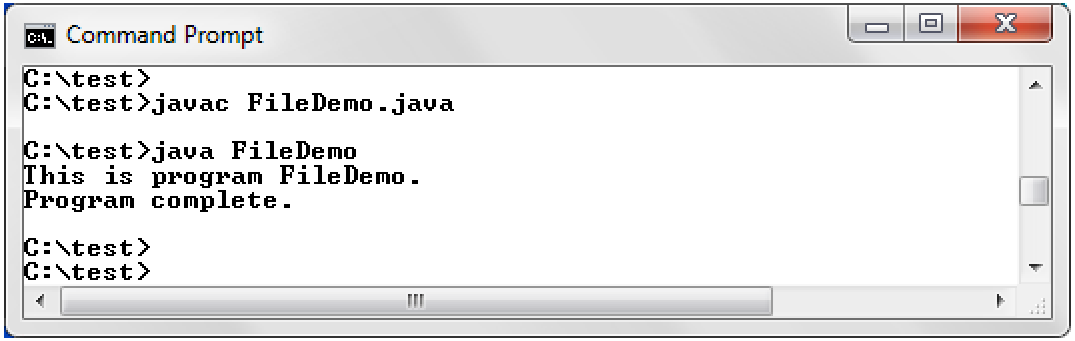


```
C:\test>
C:\test>dir
 Volume in drive C is OS
 Volume Serial Number is 046D-9AF1

 Directory of C:\test

03/01/2016  02:46 PM    <DIR>          .
03/01/2016  02:46 PM    <DIR>          ..
03/01/2016  02:44 PM                86 favorite_things.txt
03/01/2016  02:39 PM               470 FileDemo.class
03/01/2016  02:39 PM               519 FileDemo.java
               3 File(s)          1,075 bytes
               2 Dir(s)  376,772,222,976 bytes free

C:\test>type favorite_things.txt
Raindrops on roses
Whiskers on kittens
Bright copper kettles
Warm woollen mittens

C:\test>
C:\test>
```

# Add to FileDemo.java

```java
import java.util.Scanner;
import java.io.File;
import java.io.FileNotFoundException;

public class FileDemo
{
    public static void main( String[] args)
        throws FileNotFoundException
    {
        System.out.println("This is program FileDemo.");

        File favs = new File("favorite_things.txt");
        Scanner fileScanner = new Scanner(favs);

        System.out.println("Program complete.");
    }
}
```

# Compile and Run



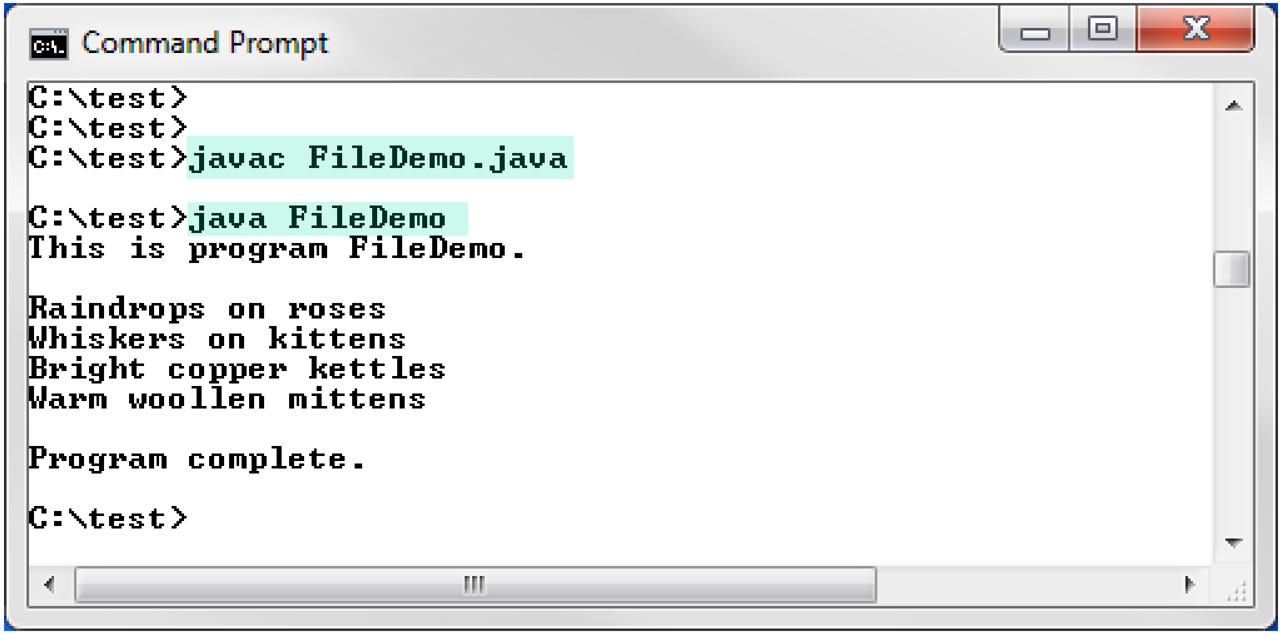
```
C:\test>
C:\test>javac FileDemo.java

C:\test>java FileDemo
This is program FileDemo.
Program complete.

C:\test>
C:\test>
```

# Add Code to Read the File

```
Scanner fileScanner = new Scanner(favs);

System.out.println();
while (fileScanner.hasNext())       // Test for end of file
{
    // Read next line from file
    String line = fileScanner.nextLine();
    System.out.println(line);
}
System.out.println();
System.out.println("Program complete.");
```

Compile and run

# FileDemo in Action

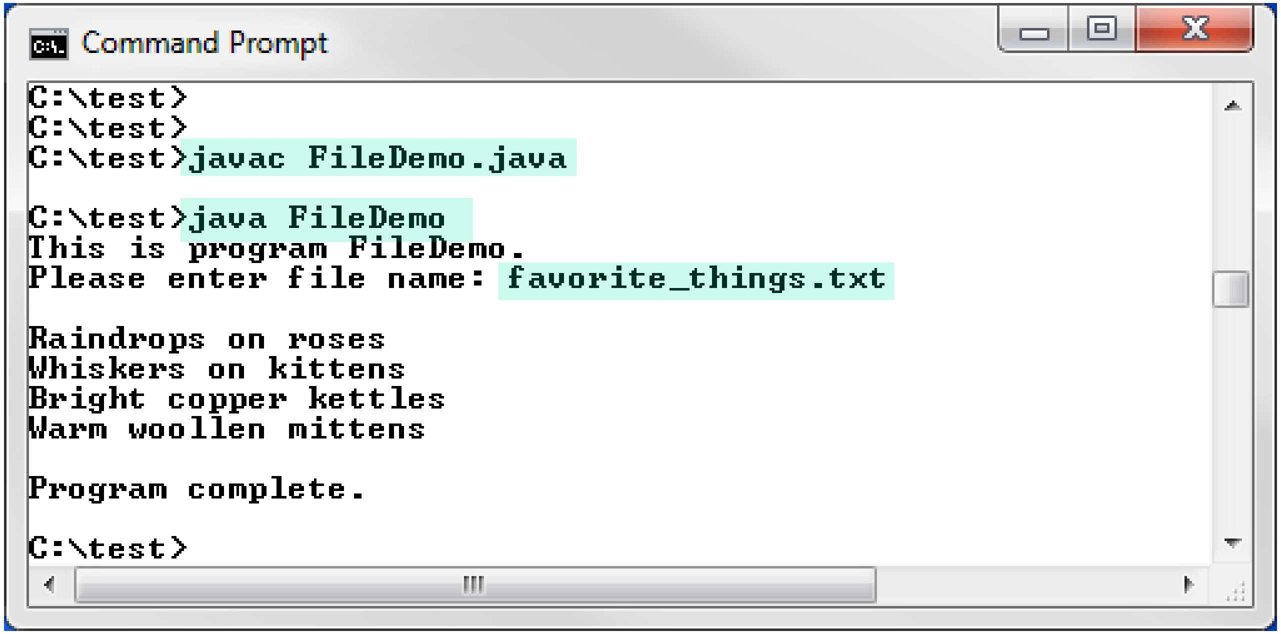


```
C:\test>
C:\test>
C:\test>javac FileDemo.java

C:\test>java FileDemo
This is program FileDemo.

Raindrops on roses
Whiskers on kittens
Bright copper kettles
Warm woollen mittens

Program complete.

C:\test>
```

# Let the user specify the file name

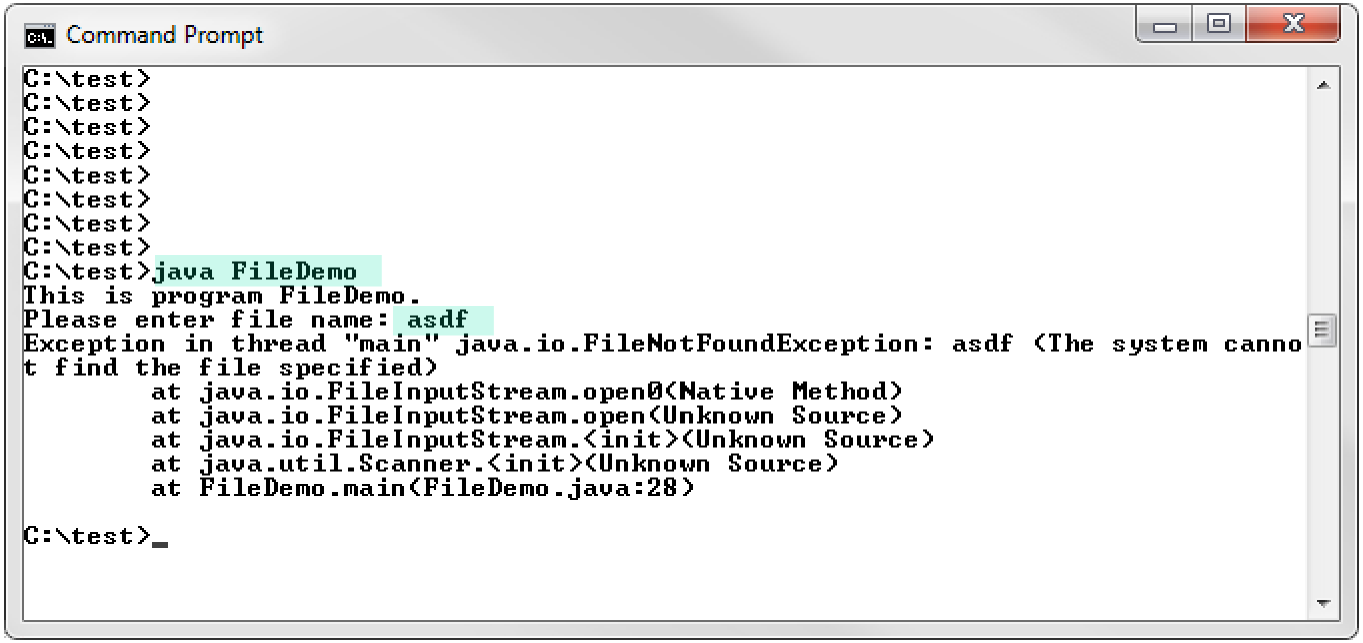
```
System.out.println("This is program FileDemo.");

// Get filename from the user
System.out.print("Please enter file name: ");
Scanner keyboardScanner = new Scanner(System.in);
String filename = keyboardScanner.nextLine();

File file = new File(filename);
Scanner fileScanner = new Scanner(file);
```

# Reading file specified by the user



```
C:\test>
C:\test>
C:\test>javac FileDemo.java

C:\test>java FileDemo
This is program FileDemo.
Please enter file name: favorite_things.txt

Raindrops on roses
Whiskers on kittens
Bright copper kettles
Warm woollen mittens

Program complete.

C:\test>
```

# What happens if file does not exist?



```
C:\test>
C:\test>
C:\test>
C:\test>
C:\test>
C:\test>
C:\test>
C:\test>
C:\test>java FileDemo
This is program FileDemo.
Please enter file name: asdf
Exception in thread "main" java.io.FileNotFoundException: asdf (The system canno
t find the file specified)
        at java.io.FileInputStream.open0(Native Method)
        at java.io.FileInputStream.open(Unknown Source)
        at java.io.FileInputStream.<init>(Unknown Source)
        at java.util.Scanner.<init>(Unknown Source)
        at FileDemo.main(FileDemo.java:28)

C:\test>_
```

# Better Error Handling

- Wrap the code that might get an error with

```
try
{
    . . .
}
```

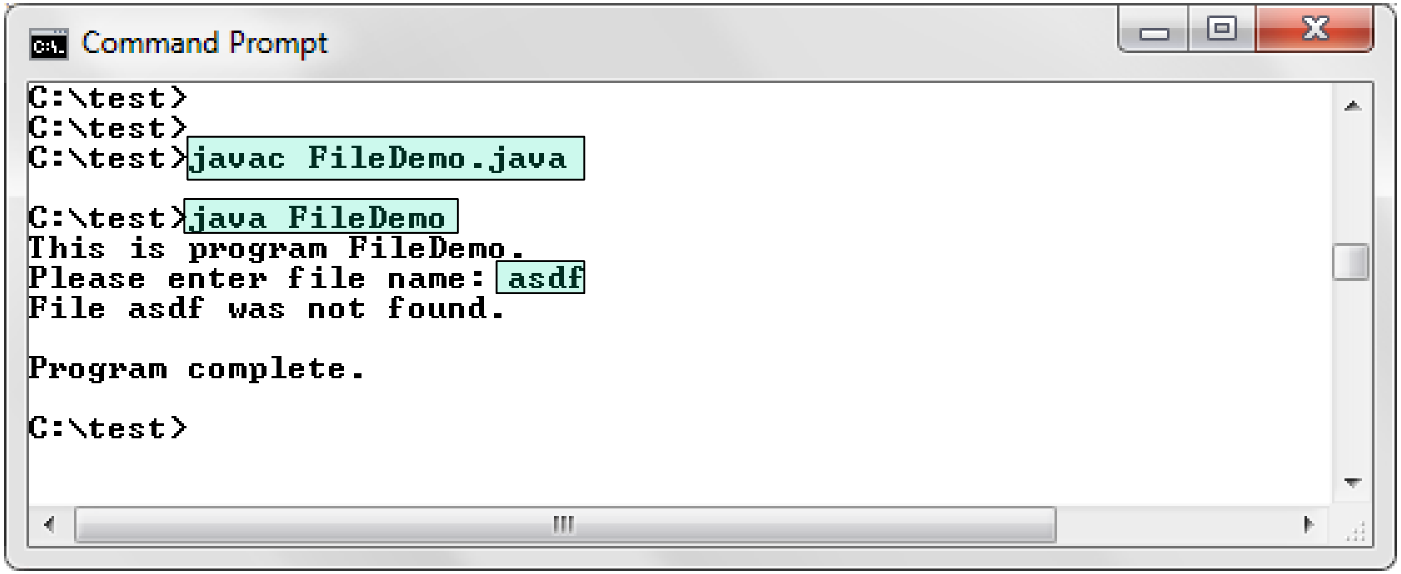
- Add a "catch" block for the exception

```
catch (exception name)
{
       . . .
}
```

# Better Error Handling

```
try
{
    File file = new File(filename);
    Scanner fileScanner = new Scanner(file);

    System.out.println();
    while (fileScanner.hasNext())      // Test for end of file
    {
        // Read next line from file
        String line = fileScanner.nextLine();
        System.out.println(line);
    }
}
catch (FileNotFoundException ex)
{
    System.out.println("File " + filename + " was not found.");
}
```

# Better Error Handling



```
C:\test>
C:\test>
C:\test>javac FileDemo.java

C:\test>java FileDemo
This is program FileDemo.
Please enter file name: asdf
File asdf was not found.

Program complete.

C:\test>
```

End of Section

# Writing Text Files

# Writing Text Files

- Similar to outputting text to the screen.
  - Methods to output various types
    - int, double, Boolean, String

- Differences
  - We have to indentify the file to be written.
  - The file might or might not already exist.

- If the file already exists
  - We can overwrite it.
  - We can append new lines to it.
    - Won't consider that for now.

# Classes for Text File Output

- FileOutputStream
  - Writes a stream of *bytes* to the file.
  - Constructor parameters:
    - filename   Name of the file to be written.
    - mode         Append?  (Set to false.)

- PrintWriter
  - Has print() and println() methods for all primitive types and for the String class.
  - Constructor takes a FileOutputStream object as a parameter.

# Steps in Writing a Text File

- Instantiate a FileOutputStream
  - Specify file name
  - Specify false for mode.  (i.e. don't append)

- Instantiate a PrintWriter object
  - Pass the FileOutputStream object to the constructor.

- Use the PrintWriter to do output to the file.
  - Same as we have done for screen output using System.out.
  - Instead of System.out, use the PrintWriter object that we have instantiated.

# When finished

- *Close* the file.

- Use the close() method of the PrintWriter object.

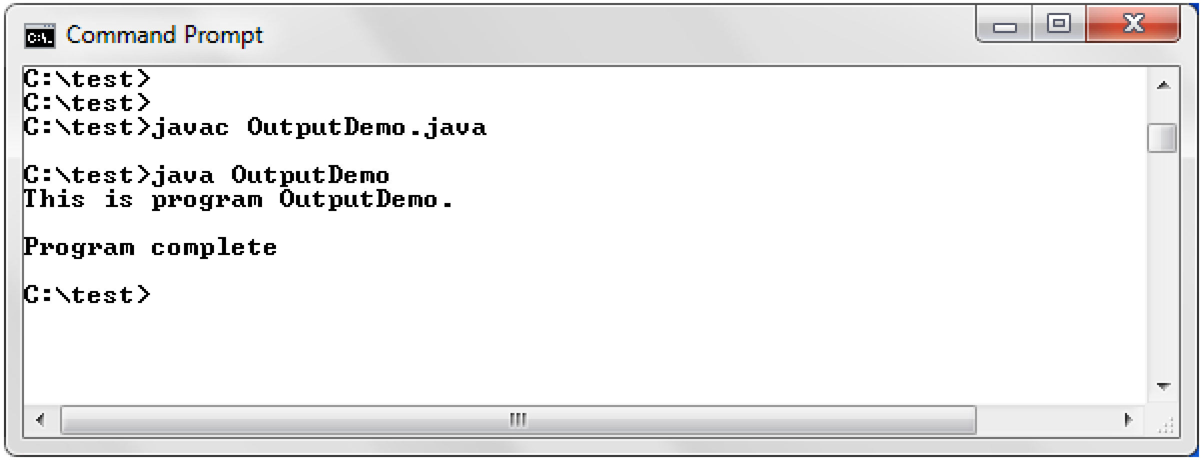
- Was not necessary for screen output.
- Necessary for files to "finish" the file.

Create a new source file, OutputDemo.java

# Initial Program

```
/*********************************************************
* Program OutputDemo.java
*
* Author: Rollins Turner
*
* This program demonstrates how to write a text file.
**********************************************************/
public class OutputDemo
{
    public static void main( String[] args)
    {
        System.out.println("This is program OutputDemo.");
        System.out.println();

        System.out.println("Program complete");
    }
}
```

# Initial Program Running

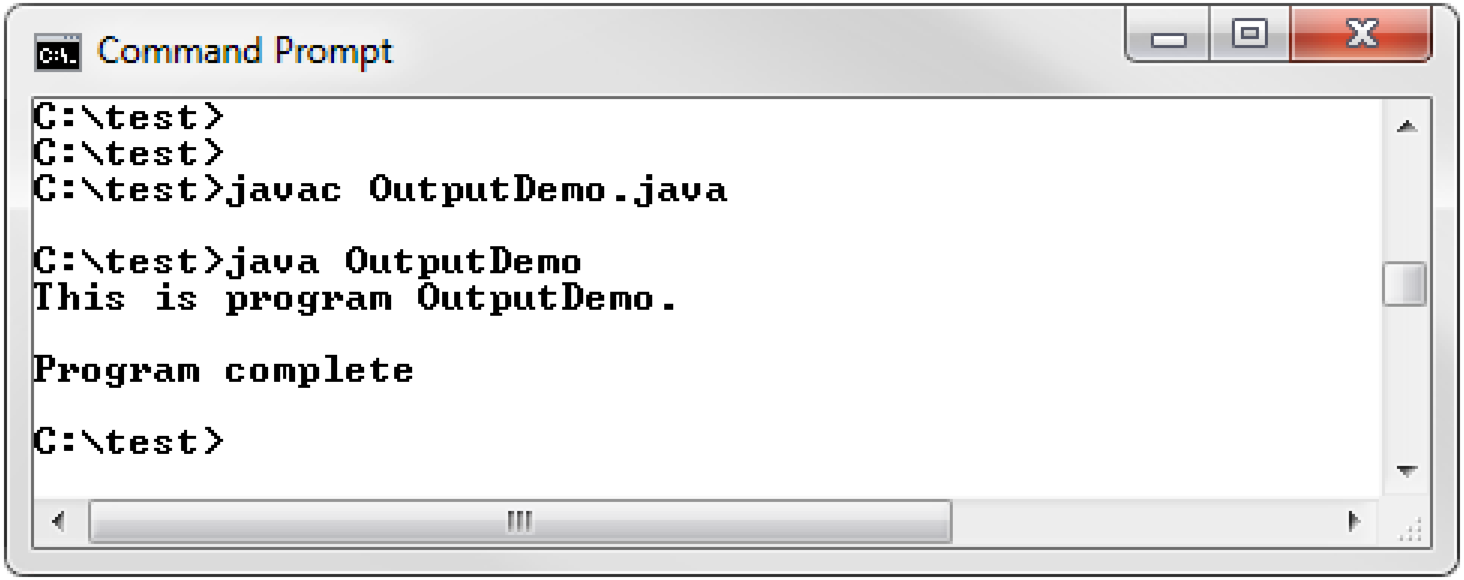


```
C:\test>
C:\test>
C:\test>javac OutputDemo.java

C:\test>java OutputDemo
This is program OutputDemo.

Program complete

C:\test>
```

# Add FileOutputStream

```
import java.io.FileOutputStream;
import java.io.FileNotFoundException;
public class OutputDemo
{
    public static void main( String[] args)
    {
        System.out.println("This is program OutputDemo.");
        System.out.println();
        try
        {
            FileOutputStream fos =
                new FileOutputStream("demo.txt", false);
        }
        catch (FileNotFoundException ex)
        {
            System.out.println("Failed to create output file.");
        }
        System.out.println("Program complete");
    }
}
```

# Compile and Run



```
C:\test>
C:\test>
C:\test>javac OutputDemo.java

C:\test>java OutputDemo
This is program OutputDemo.

Program complete

C:\test>
```

# Add PrintWriter

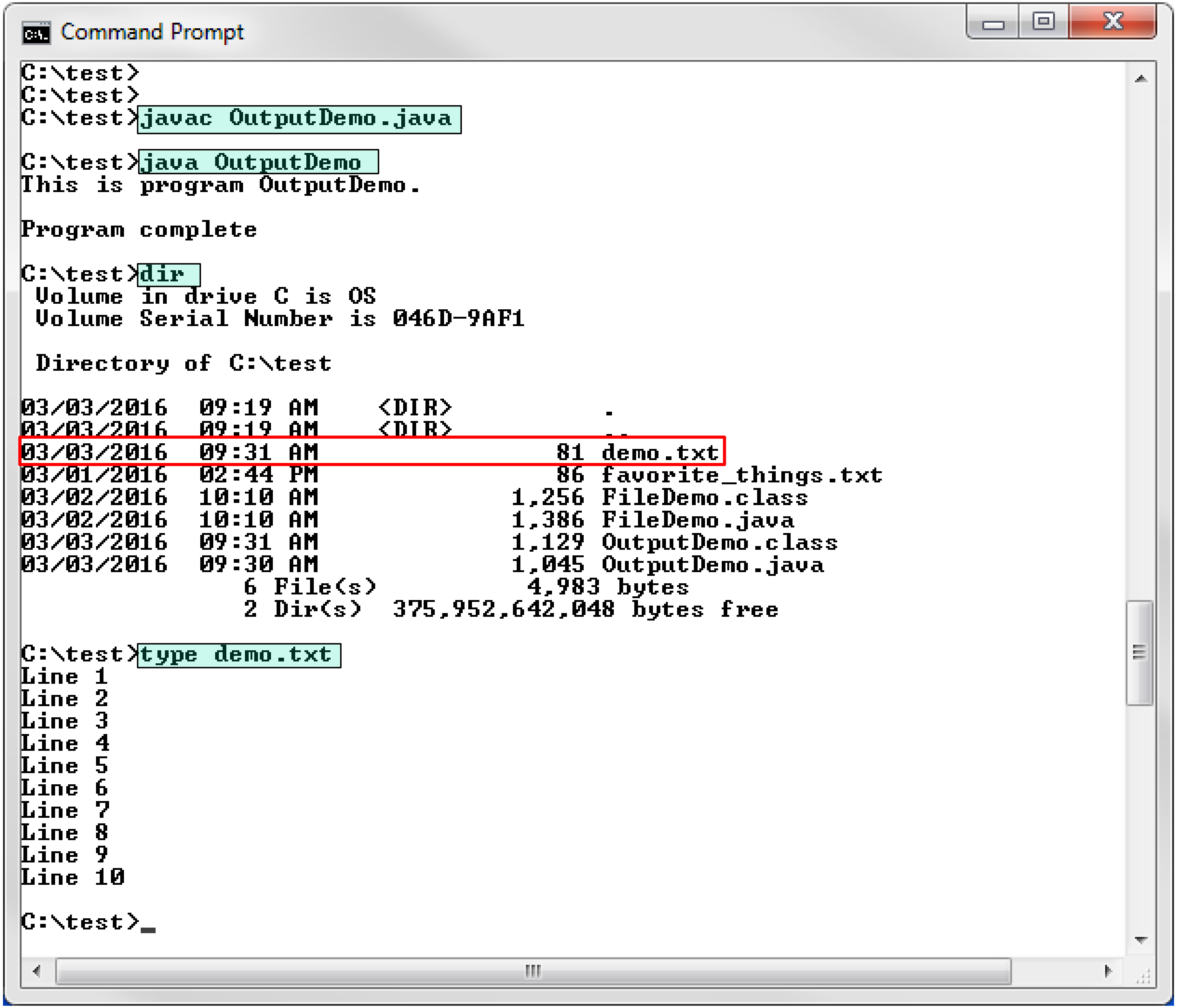
```
import java.io.PrintWriter;
public class OutputDemo
{
   ...
        FileOutputStream fos =
                new FileOutputStream("demo.txt", false);
        PrintWriter pw = new PrintWriter(fos);

        pw.close();
```

Build and run

# Add Code to Write Text

```
PrintWriter pw = new PrintWriter(fos);

for (int i = 1; i <= 10; i++)
{
    pw.println("Line " + i);
}
pw.close();
```

Build and run

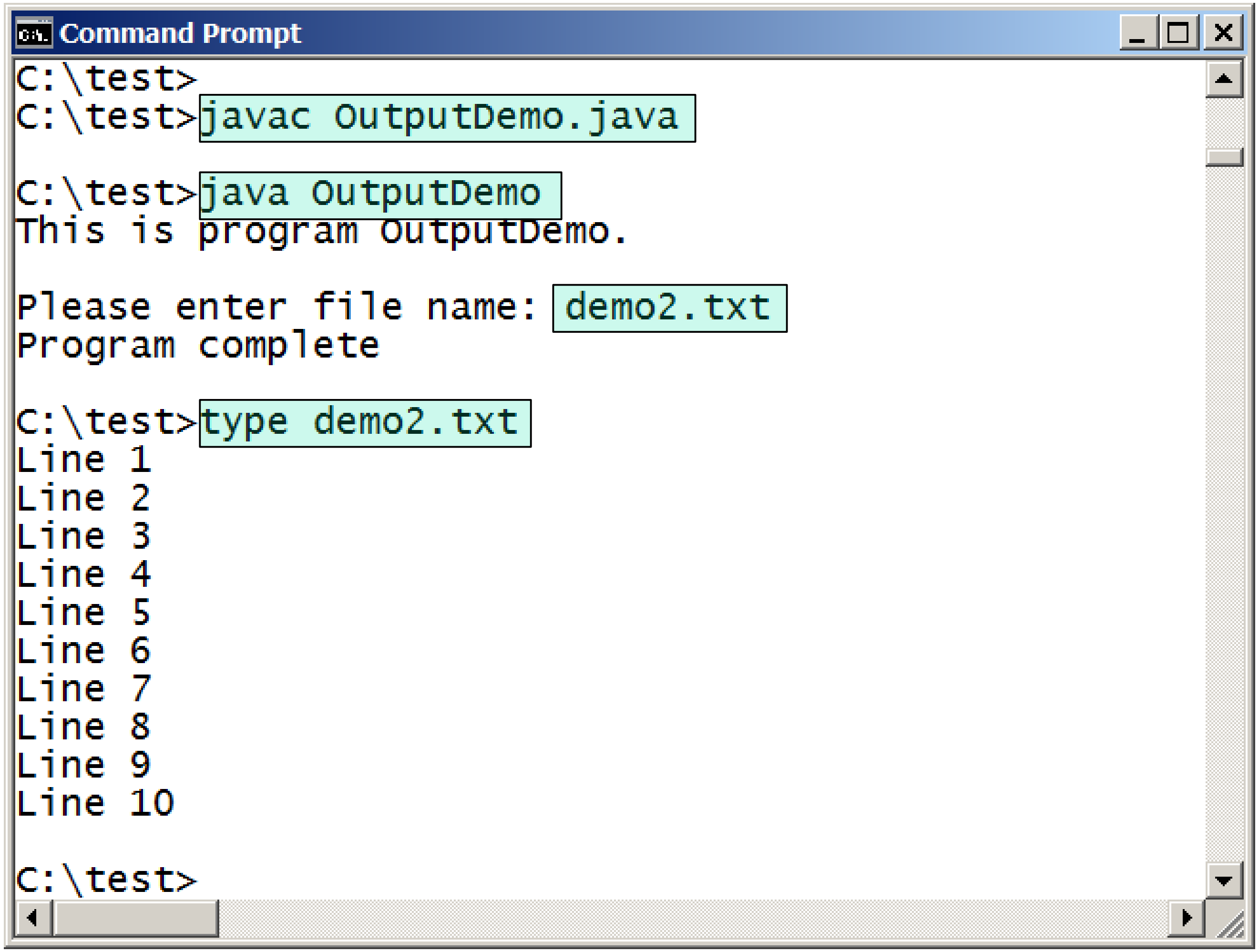


```
C:\test>
C:\test>
C:\test>javac OutputDemo.java

C:\test>java OutputDemo
This is program OutputDemo.

Program complete

C:\test>dir
 Volume in drive C is OS
 Volume Serial Number is 046D-9AF1

 Directory of C:\test

03/03/2016  09:19 AM    <DIR>          .
03/03/2016  09:19 AM    <DIR>          ..
03/03/2016  09:31 AM                81 demo.txt
03/01/2016  02:44 PM                86 favorite_things.txt
03/02/2016  10:10 AM             1,256 FileDemo.class
03/02/2016  10:10 AM             1,386 FileDemo.java
03/03/2016  09:31 AM             1,129 OutputDemo.class
03/03/2016  09:30 AM             1,045 OutputDemo.java
               6 File(s)          4,983 bytes
               2 Dir(s)  375,952,642,048 bytes free

C:\test>type demo.txt
Line 1
Line 2
Line 3
Line 4
Line 5
Line 6
Line 7
Line 8
Line 9
Line 10

C:\test>_
```

# Get file name from user

```
import java.util.Scanner;
public class OutputDemo
{
    public static void main( String[] args)
    {
        System.out.println("This is program OutputDemo.");
        System.out.println();
        try
        {
            // Get filename from the user
            System.out.print("Please enter file name: ");
            Scanner keyboardScanner = new Scanner(System.in);
            String filename = keyboardScanner.nextLine();

            FileOutputStream fos =
                new FileOutputStream(filename, false);
```

# File name specified by user



```
C:\test>
C:\test>javac OutputDemo.java

C:\test>java OutputDemo
This is program OutputDemo.

Please enter file name: demo2.txt
Program complete

C:\test>type demo2.txt
Line 1
Line 2
Line 3
Line 4
Line 5
Line 6
Line 7
Line 8
Line 9
Line 10

C:\test>
```