

Very simple file I/O

Writing to a file:

```
import java.util.Scanner;
import java.io.File;
import java.io.PrintWriter;
import java.io.IOException;

public class writeFile
{
    public static void main(String[] args) throws IOException
    {
        if (args.length != 1)
        {
            System.out.println("usage: java writeFile fileName");
            System.exit(-1);
        }

        File fw = new File(args[0]);
        PrintWriter pw = new PrintWriter(fw);

        if (!fw.exists())
        {
            System.out.println("Cannot create file: " + args[0]);
            System.exit(0);
        }

        String name = "James Gosling";
        int age = 45;
        double income = 256000.00;

        pw.println(name);
        pw.println(age);
        pw.println(income);

        Scanner input = new Scanner(System.in);
        System.out.print("Enter name: ");
        name = input.nextLine();
        pw.println(name);

        System.out.print("Enter age: ");
        age = Integer.parseInt(input.nextLine());
        pw.println(age);

        System.out.print("Enter income: ");
        income = Double.parseDouble(input.nextLine());
        pw.println(income);

        pw.close();
    }
}
```

Sample run:

```
$ java writeFile Test
Enter name: Amanda
Enter age: 19
Enter income: 12256.12
```

\$ type (or cat) Test

```
James Gosling
45
256000.0
Amanda
19
12256.12
```

Reading from a file:

```
import java.util.Scanner;
import java.io.File;
import java.io.PrintWriter;
import java.io.IOException;

public class readFile
{
    public static void main(String[] args) throws IOException
    {
        if (args.length != 1)
        {
            System.out.println("usage: java readFile fileName");
            System.exit(-1);
        }

        File fr = new File(args[0]);

        if (!fr.exists())
        {
            System.out.println("Error: file " + args[0] + " not found.\n");
            System.exit(0);
        }

        Scanner infile = new Scanner(fr);

        while (infile.hasNextLine())
        {
            String line = infile.nextLine();

            System.out.println(line);
        }

        infile.close();
    }
}
```

Sample run:

```
$ java readFile Test
James Gosling
45
256000.0
Amanda
19
12256.12
```

```
$ java readFile t
Error: file t not found.
```

Notes:

1. Please be aware of how command line processing is performed
2. The fully qualified import statements, e.g.

```
import java.util.Scanner;
not import java.util.*
```

3. How error processing is performed

What happens if the following is performed on a read only directory?

```
File fw = new File(args[0]);
PrintWriter pw = new PrintWriter(fw);
```

Answer:

```
$ java readFile t
Exception in thread "main" java.io.FileNotFoundException: t (No such file or directory)
at java.io.FileInputStream.open(Native Method)
at java.io.FileInputStream.<init>(FileInputStream.java:120)
at java.util.Scanner.<init>(Scanner.java:636)
at readFile.main(readFile.java:17)
```

3. How can this be prevented?

Answer in class...