

Java Input



imports

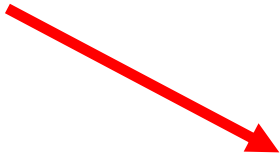
Scanner Imports

```
import java.util.Scanner;
```

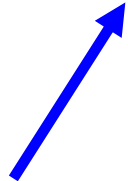
**Try to be as specific as possible
when using an import.**

Scanner Creation

reference variable



```
Scanner keyboard =  
    new Scanner(System.in);
```



object instantiation

Scanner Methods

Scanner

frequently used methods

Name	Use
nextInt()	returns the next int value
nextDouble()	returns the next double value
nextFloat()	returns the next float value
nextLong()	returns the next long value
nextByte()	returns the next byte value
nextShort()	returns the next short value
next()	returns the next one word String
nextLine()	returns the next multi word String

```
import java.util.Scanner;
```

Reading in Integers

```
Scanner keyboard =  
    new Scanner(System.in);
```

```
out.print("Enter an integer :: ");  
int num = keyboard.nextInt();
```



Reading in Integers

```
out.print("Enter an integer :: ");  
int num = keyboard.nextInt();  
out.println(num);
```

INPUT

931

OUTPUT

Enter an integer :: 931
931



Reading in Integers

reference variable

int num = keyboard.nextInt();

method call

Reading in data

```
out.print("Enter an integer :: ");
```

Prompts are used to tell the user what you want.

Open
scannerints.java

Reading in Doubles

```
Scanner keyboard =  
    new Scanner(System.in);  
  
out.print("Enter a double :: ");  
double num = keyboard.nextDouble();
```



Reading in Doubles

```
out.print("Enter a double :: ");  
double num = keyboard.nextDouble();  
out.println(num);
```

INPUT

34.33

OUTPUT

Enter a double :: 34.33
34.33



Reading in Doubles

reference variable

`double num = keyboard.nextDouble();`

method call

Open
scannerreals.java

Reading in Strings

```
Scanner keyboard =  
    new Scanner(System.in);
```

```
out.print("Enter a string :: ");  
String word = keyboard.next();
```


Reading in Strings

```
out.print("Enter a string :: ");  
String word = keyboard.next();  
out.println(word);
```

INPUT

I love java.

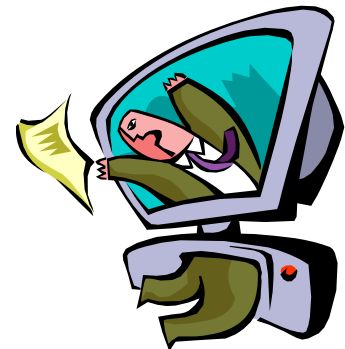
OUTPUT

Enter a string :: I love java.

I

Reading in Lines

```
Scanner keyboard =  
    new Scanner(System.in);  
  
out.print("Enter a sentence :: ");  
String sentence = keyboard.nextLine();
```



Reading in Lines

```
out.print("Enter a line :: ");  
String line = keyboard.nextLine();  
out.println(line);
```

INPUT

I love java.

OUTPUT

Enter a line :: I love java.
I love java.

Open
scannerstrings.java

nextLine() issues

```
out.print("Enter an integer :: ");  
int num = keyboard.nextInt();  
out.print("Enter a sentence :: ");  
String sentence = keyboard.nextLine();  
out.println(num + " " + sentence);
```

OUTPUT

Enter an integer :: 34
Enter a sentence :: 34

INPUT

34
picks up \n

nextLine() picks up whitespace.

nextLine() issues

```
out.print("Enter an integer :: ");  
int num = keyboard.nextInt();  
keyboard.nextLine();    //pick up whitespace  
out.print("Enter a sentence :: ");  
String sentence = keyboard.nextLine();  
out.println(num + " " + sentence);
```

OUTPUT

Enter an integer :: 34
Enter a sentence :: picks up \n
34 picks up \n

INPUT

34
picks up \n

nextLine() picks up whitespace.

Open
nextlineissues.java

Multiple Inputs

INPUT

1 2 3 4 5

```
Scanner keyboard =  
    new Scanner(System.in);
```

```
out.println(keyboard.nextInt());  
out.println(keyboard.nextInt());  
out.println(keyboard.nextInt());
```

OUTPUT

1
2
3

Open multithread.java

Old School Input

```
BufferedReader keyboard =  
    new BufferedReader(  
        new InputStreamReader( System.in ) );  
  
System.out.print("Enter a word :: ");  
String s = keyboard.readLine();  
  
System.out.println(s + '\n' );
```

Old School Input

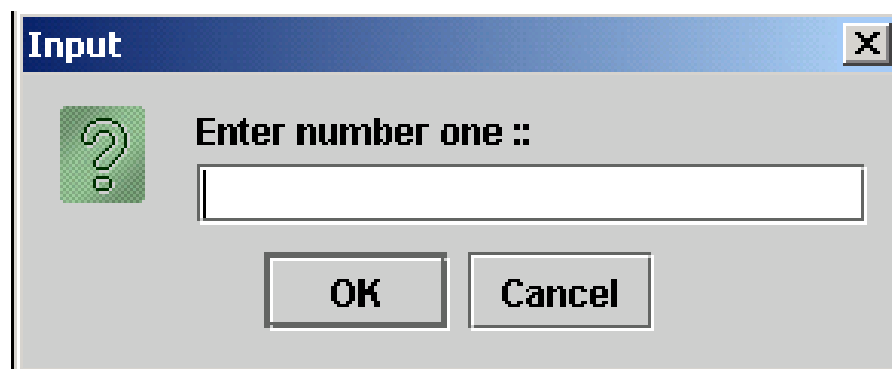
`readLine()` reads in all data as text / string data.
The text you read in must be converted over to the appropriate type before it can be stored.

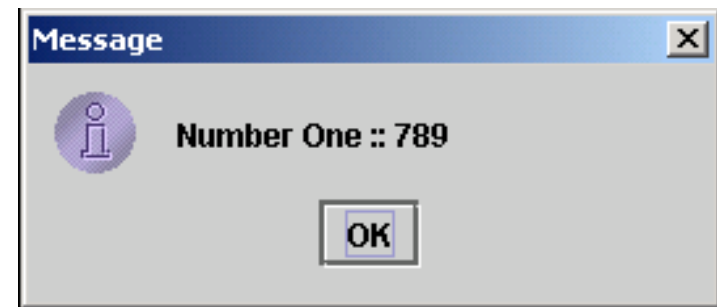
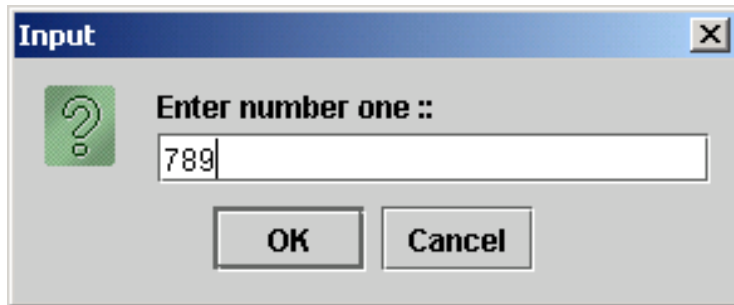
```
System.out.print("Enter an integer :: ");  
one = Integer.parseInt(keyboard.readLine());
```

```
System.out.print("Enter a double :: ");  
two = Double.parseDouble(keyboard.readLine());
```

Open
oldschoolone.java
oldschooltwo.java

GUI





//GUI INPUT BOX

```
input= JOptionPane.showInputDialog("Enter an integer :: ");  
one = Integer.parseInt(input);
```

//GUI OUTPUT BOX

```
JOptionPane.showMessageDialog(null, "Integer value :: " + one);
```

Open guihelp.java

**Start work
on the labs**