

CSCI 1320 Java Commands

Hello, World!

```
1 // imports go here
2 public class Hello {
3
4     public static void main(String [] args) {
5         System.out.println("Hello, World!"); // yay!
6     }
7
8     /*
9     additional methods go here
10    ... after main() but inside the class
11    ... but not inside a comment!
12    */
13 }
```

Expected output (on screen):

Hello, World!

Declaring Variables

```
1 float x;
2 double y,z;
3 int ans;
4 char let;
5 String name;
6 boolean isFun;
```

x	?	ans	?
y	?	let	?
z	?	name	?
isFun	?		

Initializing Variables

```
1 y= 0.1;
2 z= 7.5 + y;
3 name= "Fred"; // String use double quotes
4 let= 'R'; // chars use single quotes
5 a= (int) x;
6 a= a + 1;
7 isFun= true;
```

y	0.1	name	"Fred"
z	7.6	isFun	true
a	8		

Displaying Values

```
1 System.out.println("Hello");
2 System.out.print("Answer is: ");
3 System.out.println(a);
4 System.out.println("Name is: "+name);
```

Output of statements assuming variables have values from previous examples:

Hello
Answer is: 8
Name is: Fred

Getting Input

```

1 import java.util.Scanner; // at top of file
2 . . .
3 Scanner kb;
4 kb= new Scanner(System.in);
5 int a;
6 double x;
7 String str1,str2;
8
9 System.out.print("Enter age: ");
10 a= kb.nextInt();
11 kb.nextLine(); // toss out extra newline
12
13 System.out.print("Enter num: ");
14 x= kb.nextDouble();
15 kb.nextLine(); // toss out extra newline
16
17 System.out.print("Enter word: ");
18 str1= kb.next();
19 kb.nextLine(); // toss out extra newline
20
21 System.out.print("Enter sentence: ");
22 str2= kb.nextLine();
23 // .nextLine() tosses out extra newline

```

a	?
x	?
str1	?
str2	?

Enter age: 7
Enter num: 2.5
Enter word: hi
Enter sentence: How are you?

a	7
x	2.5
str1	"hi"
str2	"How are you?"

Integer Arithmetic

```

1 int a,b,c,d,e;
2 a= 7;
3 b= -1;
4 c= a + b * 3; // 4
5 d= b - a / 3; // -3
6 e= a % 3;    // 1

```

a	7
b	-1
c	4
d	-3
e	1

Floating Point Arithmetic

```

1 double a,b,c,d;
2 a= 4.0;
3 b= -2.2;
4 c= a + b * 3;          // -2.6
5 d= Math.sqrt(a) + b; // -0.2

```

a	4.0
b	-2.2
c	-2.6
d	-0.2

Strings

```

1  int num1,num2;
2  String one, two, three;
3  one= "Amy";
4  two= "Hi, there!";
5  if (one == "Amy") { // nope!
6      System.out.println("Greetings!");
7  }
8  if (one.equals("Amy")) { // yep!
9      System.out.println("Hi, Amy!");
10 }
11 num1= one.length();
12 System.out.println(one.charAt(2));
13 num2= one.indexOf("the");
14 System.out.println(one.toUpperCase());
15 three= two.substring(4,7)+"n";

```

num1	3
num2	4
one	"Amy"
two	"Hi, there!"
three	"then"

Expected output:

```

Hi, Amy!
y
FRED

```

If Statements

```

1  if (a < 0) {
2      // do this if true
3  }
4  else {
5      // do this if false
6  }

```

Boolean Expressions

```

1  int a,b;
2  boolean c,d,e,f,g;
3  a= 7;
4  b= -1;
5  c= true;
6  d= a<b;
7  e= a<b || a>0;
8  f= a<=b && a!=0;
9  g= !isCool && a==7;

```

a	7
b	-1
c	true
d	false
e	true
f	false
g	true

While Loops

```

1  while (a < 0) {
2      // loop body
3  }
4
5  do {
6      // loop body
7  } while (a < 0);

```

For Loops

```

1 int i;
2 for (i=0; i<5; i=i+1) {
3     System.out.print(""+i+",");
4 }

```

Expected output:

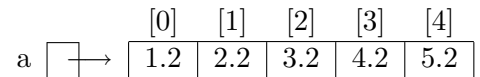
0,1,2,3,4,

Arrays

```

1 int i;
2 double [] a; // declare array name
3 a= new double[5]; // reserve 10 doubles
4 a[0]= 1.2;
5 for (i=1; i<5; i++) {
6     a[i]= a[0]+i;
7 }

```



Methods

```

1 public static void showSum(double a, double b) {
2     double c;
3     c= a + b;
4     System.out.println("Sum is: "+c);
5 }
6
7 public static double calcSum(double a, double b) {
8     double c;
9     c= a + b;
10    return c;
11 }
12
13 // To call these methods:
14 double x= 7.5;
15 showSum(x,1.0);
16 showSum(x*2.0,x-2.0);
17 x= calcSum(1.0,2.2); // x= 3.2
18 System.out.println(x);

```

Expected output:

Sum is: 8.5
Sum is: 20.5
3.2

Formatted Output

```

1 double x= 7.5;
2 String str= "Fred";
3 int a= 2;
4
5 System.out.printf("%s is %d\n",str,a);
6 System.out.printf("A%5d %4.2fB\n",str,a);
7 System.out.printf("A%-5d %4.2fB\n",str,a);

```

Expected output:

Fred is 2
A 2 7.50B
A2 7.50B

Reading Text Files: Scanner

```
1 // requires some imports:
2 import java.util.Scanner;
3 import java.io.FileNotFoundException;
4 import java.io.FileInputStream;
5
6 public static double sumFile(String filename) throws
7     FileNotFoundException {
8     Scanner numFile;
9     double num,total=0.0;
10
11     numFile= new Scanner(new FileInputStream(filename));
12     while (numFile.hasNextLine()) {
13         num= numFile.nextDouble();
14         total= total + num;
15         numFile.nextLine(); // toss out \n
16     }
17     numFile.close();
18     return total;
19 }
```

Suppose datafile looks like this:

1.5
10.2
2.3
5.2

Expected return value: 19.2

Reading Text Files: BufferedReader

```
1 // requires some imports:
2 import java.io.BufferedReader;
3 import java.io.FileReader;
4 import java.io.IOException;
5
6 public static double sumFile(String filename) throws
7     IOException {
8     BufferedReader numFile;
9     double total=0.0;
10    String str;
11
12    numFile= new BufferedReader(new FileReader(filename));
13    while ((str= numFile.readLine())!=null) {
14        total= total + Double.parseDouble(str);
15    }
16    numFile.close();
17    return total;
18 }
```

Suppose datafile looks like this:

1.5
10.2
2.3
5.2

Expected return value: 19.2

Saving Text Files

```
1 // requires some imports:
2 import java.io.FileNotFoundException;
3 import java.io.PrintStream;
4
5 public static void saveData(String filename) throws
6     FileNotFoundException
7 {
8     PrintStream f= new PrintStream(filename);
9     double y= 3.8234;
10
11     f.print("This is so fun!");
12     f.print("How are you?\n");
13     f.println("What do we do now?");
14     f.printf("I am am %8.1f feet tall\n",y);
15     f.close();
16 }
```

Expected output (in junk.txt):

```
This is so fun!How are you?
What do we do now?
I am    3.8 feet tall
```

Try-Catch

```
1
2 int num;
3 Scanner kb= new Scanner(System.in);
4
5 try {
6     System.out.print("Enter number: ");
7     num= kb.nextInt();
8     System.out.println("You entered: "+num);
9 }
10 catch (InputMismatchException e) {
11     System.out.println("Must enter an integer");
12     System.out.println(e);
13 }
```

Expected output if user enters 7.

```
Enter number: 7
You entered: 7
```

Expected output if user enters
"frog".

```
Enter number: frog
Must be an integer
java.util.InputMismatchException
```
