Java Reference Card

1. Classes

The following is an example of a "main" class:

```
public class Calculator {
    public static void main(String[] args) {
    }
}
```

and the following is an example of a "utility" class (with no methods):

```
public class Geometry {
}
```

2. Methods

The following is an example of a method declaration with an empty body:

```
public static double circleArea(double radius) {
}
```

and the following is an example of an invocation of this method (assuming that it is in the Geometry class):

```
    area = Geometry.circleArea(radius);
```

3. Conditionals

The following is an example of an `if` statement with an `else` clause:

```
    if (price > 100.00) {
        discount = 0.40;
    } else {
        discount = 0.10;
    }
```

4. Operators

Arithmetic Operators		Logical Operators			Relational Operators	
Addition	+	And	&	&&	Equal	==
Decrement	−	Excl. Or	^		Greater than	>
Division	/	Incl. Or	|	||	Greater than or equal	>=
Increment	++	Not	!		Less Than	<
Int. Division	/				Less than or equal	<=
Multiplication	*				Not equal	!=
Modulus	%					
Negation	−					
Subtraction	−					

4. Type Conversion

Example Expression	Type	Value
`(1 + 2 + 3 + 4) / 4.0`	`double`	`2.5`
`"1234" + 99`	`String`	`"123499"`
`11 * 0.25`	`double`	`2.75`
`(int)2.71828`	`int`	`2`
`(int)11 * 0.25`	`double`	`2.75`
`11 * (int)0.25`	`int`	`0`
`(int)(11 * 0.25)`	`int`	`2`

5. Library Methods

Signature	Purpose
`Math.abs(double v)`	Absolute value
`Math.cos(double a)`	Cosine
`Math.max(double x, double y)`	Maximum
`Math.min(double x, double y)`	Minimum
`Math.pow(double v, double p)`	`v` raised to the `p` power
`Math.sin(double a)`	Since
`Math.sqrt(double v)`	Square root
`Math.tan(double a)`	Tangent
`Math.toDegrees(double r)`	Radians to degrees
`Math.toRadians(double d)`	Degrees to radians
`Math.E`	The base of the natural log
`Math.PI`	The circumference over the radius

5. Input

Input Using a `Scanner` Object

```
import java.util.Scanner;

double d;
int i;
Scanner in;
String s;
in = new Scanner(System.in);
d = in.nextDouble();
i = in.nextInt();
s = in.nextLine();
```

Input Using the `JMUConsole` Class

```
double d;
int i;
String s;

JMUConsole.open();
d = JMUConsole.readDouble();
i = JMUConsole.readInt();
s = JMUConsole.readLine();
JMUConsole.close();
```

6. Output

Both the `System.out` object and the `JMUConsole` class have the following methods. (Recall that `JMUConosole.open()` must be called before it can be used for either input or output and `JMUConsole.close()` should be called just before the program terminates.)

`print()`	Can be passed a `double`, `int`, or `String`
`println()`	Can be passed a `double`, `int`, or `String`
`printf()`	Is passed a format string and one value for each format specifier

Example Specifier	**Description**
`%d`	Integer
`%5d`	Integer in a field of width 5
`%f`	Floating-point
`%f5.2`	Floating-point in a field of width 5 with 2 places to the right of the .
`%s`	String

Complete Example **01234567891023456789012345678 90**

```
printf("%2d%5.2f", 5, 8.1)        5 8.10
printf("%10d%8.4f", 5, 8.1)                 5  8.1000
```