Java Reference Card

1. Classes and Methods

The following is an example of a class (Calculator) with a single method "main":

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

The following is an example of a method declaration with an empty body within a class named Geometry:

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

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

```
     double radius = 5.0;
     area = Geometry.circleArea(radius);
```

3. Operators

Arithmetic	Operators
Addition	+
Decrement	--
Division	/
Increment	++
Int. Division	/
Multiplication	*
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. Math Library Methods/Constants

Signature	Purpose	Return type
`static double Math.abs(double v)`	Absolute value	Double
`static double Math.cos(double a)`	Cosine	Double
`Math.pow(double v, double p)`	`v` raised to the `p` power	double
`Math.PI`	The constant for π	NA

6. Arrays

The following is an example of an array declaration:

```
double[] grades;
```

This example allocates memory for an array:

```
grades = new double[3];
```

The following examples uses elements of an array (assumes variables have been declared earlier):

```
grades[2] = 85.5;
finalGrade = grades[2];
```

7. Loops

Examples assume variables are previously declared.

For loop:	`For (i = 0; i < n; i++) {` `}`
While loop	`While (i < n) {` `}`
do while loop	`do {` `} while (i < n);`

8. 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();
```

9. Output

The `System.out` object has the following methods:

`print()`	Can be passed a `double`, `int`, or `String`
`println()`	Can be passed a `double`, `int`, or `String` and includes a newline at the end
`printf()`	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
`%5.2f`	Floating-point in a field of width 5 with 2 places to the right of the decimal point .
`%s`	String

Printf Complete Example **01234567891023456789012345678 90**

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