Name ____ Period __________

Variable Types

Your Tasks (Mark these off as you go)

- □ Declare a class
- □ Declare and initialize a variable
- □ Indicate the four primitive variable types encountered in this class
- □ Create a String variable type
- □ Receive credit for this lab guide

□ Declare a java driver class

The main method, often referred to as the “driver” is required for all java programs. A “driver class” is the class which contains this method. In the space write a driver a class called “VariableTypes”.

□ Declare and initialize a variable

A variable is a name for a location in memory used to hold a data value. A variable declaration instructs the compiler to reserve a portion of main memory space large enough to hold a particular type of value, and it indicates the name by which we refer to that location.

Below is an example of how to declare and initialize an *int* type variable. Notice in the example, that the variable, *total*, was declared and initialized on the same line. Recall however, that initializing a variable at the time of declaration is optional.

type

optimal initialization

```
int total = 50;
```

variable name

A variable declaration can have multiple variables of the same type declared on one line. Each variable on the line can be declared with or without an initializing value. Below is an example, type   initialized

```
int count, minimum = 0, result;
```

not initialized

Write code that could be used to

(a) Declare and initialize three *int* type variables (a, b, c) all on the same line.

(b) Declare a fourth variable called sum and assign the sum of the variables a, b, and c to this value

(c) Print the sum to the console

□ Indicate the four primitive variable types encountered in this class

As we previously learned, there are 8 different primitive data types in the Java language. For the purposes of this class, you only need to be concerned with the following primitive variable types

Variable type	**Description**
int	Stores whole numbers from -2,147,483,648 to 2,147,483,647
double	Stores fractional numbers. Sufficient for storing 15 decimal digits
boolean	Stores true or false values
char	Stores a single character/letter or ASCII values

Refer to the description of each of the four primitive data types above. For each value below, declare an appropriate variable and initialize the variable to the value.

(a) 2.0

(b) -86

(c) False

(d) 0

□ Convert between data types

It is sometimes necessary to convert a data value of one type to another. But, we must be careful that we don't lose important information in the process. A conversion between one primitive data type to another falls into one of two categories: widening conversions and narrowing conversions. Widening conversions are the safest because they usually do not lose information. They are called widening conversions because they go from one data type to another type that uses an equal or greater amount of space to store the value. Below is an example,

```
int i = 2;

double d = i; Widens variable i to 2.0

System.out.println(i);//prints 2.0
```

Narrowing conversions are more likely to lose information than widening conversions are and are therefore not permitted in Java without explicitly telling Java to do so. The following code snippet for example would result in an error,

```
              Assigns 2.0 to variable d
double d = 2;

int i = d;  Not allowed because the trailing zero is
            lost
```

Narrowing in Java can only be done through *casting.* In the above example, *d* can be casted to an *int* as follows,

```
double d = 2;       used to cast d to an int data
                    type
int i = (int) d;
```

For each code segment, indicate whether its legal or illegal. Then provide an explanation.		
	legal/illegal	**Explanation**

`int i = 5;`
`double d = i;`		

`double p = 3.14;`
`int r = p;`		

`double age = 16;`
`int g = (int) age;`		

`int lightSpeed = 3.0E8;`		

□ Create a String variable type

In Java a variable name represents either a primitive or an object. Consider the following two declarations,

```
int num;
String name;
```

The first declaration creates a variable that holds an integer value, as we have seen before. The second declaration creates a String variable that holds a reference to String object (more on this later). String literals are defined using quotations marks as follows,

```
name = “Java is awesome!”;
```

Submit this portion of the lab to Pluska to receive credit for the lab guide. Once received, your completed code challenges will also be graded and will count towards your final lab grade.

Write a single line of code that declares a variable and assigns your name to it. Write a second line of code that prints the variable storing referencing your name to the console.

□ Receive Credit for this lab guide

Submit this portion of the lab to Pluska to receive credit for the lab guide. Once received, your completed code challenges will also be graded and will count towards your final lab grade.

Submit this portion of the lab to