An Overview of Chapter 2 (Java Fundamentals)

What is a byte in computer memory?
How are bytes located (found/addressed)?
In a Java program there are primitive data types including the following for numeric data:

<table>
<thead>
<tr>
<th>DATA TYPE</th>
<th>SIZE</th>
<th>RANGE</th>
</tr>
</thead>
<tbody>
<tr>
<td>byte</td>
<td>1 byte</td>
<td>Integers in range -128 to +127</td>
</tr>
<tr>
<td>short</td>
<td>2 bytes</td>
<td>Integers in range -32,768 to 32,767</td>
</tr>
<tr>
<td>int</td>
<td>4 bytes</td>
<td>Integers in range -2,147,483,648 to 2,147,483,467</td>
</tr>
<tr>
<td>long</td>
<td>8 bytes</td>
<td>Integers in range (look it up)</td>
</tr>
<tr>
<td>float</td>
<td>4 bytes</td>
<td>Floating point numbers from ±3.4 X 10^{-38} to ± 3.4 X 10^{38}. (7 digits of accuracy).</td>
</tr>
<tr>
<td>double</td>
<td>8 bytes</td>
<td>Floating point number from ±1.7 X 10^{-308} to ±1.7 X 10^{308}. (15 digits of accuracy)</td>
</tr>
</tbody>
</table>

Java has a set of pre-defined key words (sometimes called reserved words). These have special meaning in compilation, and cannot be redefined or used in a manner other than their original purpose. The words (byte, short, int, long, float, double) appearing in the table above are all key words. Of course, there are many others.

In Java items in memory may be identified using “variables”, which is an identifier that can be constructed by the programmer using the following rules:

- The first character must be a letter a-z or A-Z, an underscore ( _ ) or a dollar sign ( $ ).
- After the first character, the remaining may consist of those listed above plus digits ( 0 – 9 ).
- Java is a case sensitive, so letters a – z are different than A – Z when used in a variable name.
- Identifiers cannot include spaces.
- User defined identifiers cannot be key words.

General rules of thumb to use when creating variable names:

- Choose a name that represents the quantity that it identifies. (We’ll look at some examples).
- Normally variable names begin with a lowercase letter. When a name contains other words, successive words begin with a capital letter. Examples: monthlyInterest; rateOfSpeed; nextPayment. This is a convention to follow, but is not syntactically required.
**Arithmetic Operators:** +, -, *, /, % (We will discuss these) As you consider these make sure you understand what happens when dividing integers versus floating point numbers.

**Order of Precedence of Operators.** Which operations are performed first? How can you override the precedence?

In Java all variables must be declared prior to use. The declaration determines the type that the variable will be. Consider the following:

- `int  numberOfMonths;`
- `short age;`
- `long nationalDebt;`
- `float intRate1 = 6.0f; //why is the f included here?`
- `double interestRate2 = 3.5;`

Java has a class called String, which is used to declare and use strings. Think of a string as a sequence of characters, but remember that a string is NOT a primitive type, rather it is an “object” in the String class. Consider the following declarations:

- `String name1;`
- `String name2 = “Elmer Fudd”;`
- `String cityState = “Kalamazoo, Michigan”;`
- `String allLetters1 = “The quick brown fox ”;`
- `String allLetters2 = “jumped over the lazy dogs.”;`
- `String allLetters = allLetters1 + allLetters2;`

Can you figure out from these examples what double quotes do?

Is String a key word?

**Primitive Type versus Object**
At this point we will say a few words about the difference in primitive types and objects in some class. This will involve “value” and “reference”.

**Literals versus Variables**
We saw above that variables are used to identify memory locations allocated in a program to hold information. Literals are the constants used within the program to compute values and/or assign values to primitives or to create objects.

What are the literals in the examples provided in the preceding material?