Review

8 primitive types- integer, double, char, Boolean

String in non-primitive type

Primitive types copy the value

Non primitive types copies address value

String b+ new String (“Learning Java”);

**Always creates a new string object**

**String value or length can never be changed**

**PACKAGES**

Class library is a collection of that classes we can use

Java standard class library

Standard class library is organized into packages

Java.lang java.util. java.swing java.awt are examples of packages

Want to use a class from a package

***import*** java.util.Scanner;

want to import all of package

***import*** java.util.\*;

all of java.lang are automatically imported

**Math Class**

Part of the java.lang package

Contains various mathematical functions that include absolute value, square root and exponents

Methods of math class are static methods

Static methods can be invoked through the class name- no object of math class is needed.

ex

Value = Math.cos(90) + math.sqrt(delta);

**Imprecision of float point**

Not real number

Finite number that have max and min values

Subject to round off errors

Formatting output

Often necessary to format value to display properly

Part of java.tet package

NumberFormat class can format values as currency or percentage

decimalFormat class format values base on decimal

Number Format

Java.text.NumberFormat;

Has static method – returns a formatter object

Each formatter object has a method called a format that returns a string with the specified information in the appropriate format

String format (double Number)

Returns a string containing the specified format

Decimal Format

Used to format a floating point value

Constructor of the decimal format class takes a string that represents a pattern for the formatted number

**FLOW of CONTROL**

Order of statement execution

Linear method- one after another

If-statement

If is a java reserved word--- if(condition) statement;

Condition must be a Boolean type (True of False)

Boolean expressions- condition uses one of Java’s equality operator

== equal to if(a==b) not to be confused with (a=b)

!= not equal to

< less than

> greater than

<= less than or equal to

>= greater than or equal to

**Comparing String**

A string is an object

Equals can be called to check if stings are the same all the way down to character order and uppercase/lowercase

Equals method returns a Boolean result

If (name1.equals(name2)

System.out.println(“Same name”)

Else

System.out.println(“Not same”)

**If-else statement**