2 categories of data types

Primitive- just contains a value

Int, double, float, byte, long, short, Boolean, char

Classes- data types that have both spaces to hold data and functions for working with that data

Random rnd;

Rnd = new Random(); memory is borrowed from system heap (free store)

New tells Java that I want the memory to hold the brand new object

After the new we see the constructor for the class being called

The constructor initializes the values of all variables inside object

Constructor has the same name as the name of the class and may take in values to initialize variables

The string is a class. Any string variable is a string object.

Strings are easier to create than any other object

String firstName;

firstName = “Morgan”; or firstName = new String(“Morgan”);

classes are stored in packages

package == library

full name of any class … package.name.ClassName

java.lang.string

the default package is java.lang

every other package must be imported

import java.util.Random;

import java.util.\*; //gives access to everything in java.util

Scanner – java.util.Scanner

Import java.util.Scanner;

Scanner sc = new Scanner(System.in);

sc.next() 🡪 return the next sequence of chars as a string up to the next whitespace(tab, space, eol)

sc.nextInt() 🡪 same as next() excepts converts to an int

sc.nextDouble()

sc.nextFloat()

sc.nextLine() 🡪reads everything up through next eol and then stops. Always stops reading after sucking in the eol marker

if (str1.equals(str2)){

}

if (str1.equalsIgnoreCase(str2)){

}

Javadoc – java’s automated documentation system

/\*\*

Lines of explanation

@author Morgan

@param a first number

@param b second number

@return int that represents the sum

\*/

Javadoc comments should appear immediately before a class as well as each public function you think needs commenting

Sequence

Selection

Repetition

2 types of repetition

\*counter-controlled: ahead of time know how many times to repeat steps

for

\*sentinel-controlled: can’t predict when you’re going to stop

While

Do…while

For loop: counter-controlled

For (counter variable declaration and initialization; test on the variable; adjustment to counter variable ){

}

for(int i = 0; i < 10; i = i + 1 or i++){

}

Sentinel-controlled loops:

While – preforms the test at beginning of the loop-chance that body of the loop never executes

Do… while – preforms its test at the end-body of the loop will execute at least once

While (Boolean expression){

Stuff to repeat

Adjust the Booleans expression variables

}

Do{

}while (Boolean expression)

SHAPE CALC  
do{ if

Else if

Else if

}while (!choice.equals(“Q”));