Creating objects

The new operator is used to create an object

Title = new String(“Java software solutions”);

This calls the string constructor, which is a special method that sets up the object.

Invoking Methods

We’ve seen that once an object has been instantiated, we can use the dot operator to invoke (call) it’s methods

Assignment Revisited

The act of assignment takes a copy of a value and stores it in a variable

For object references, assignment copies the address

Before: name1 = steve jobs

Name 2 = steve wozniak

Name 2 = name1;

After: name1 = “steve jobs”

Name 2 = “steve jobs”

Steve wozniak is still there but name 2 redirects to name one

Aliases and Garbage Collections

Two or more references that refer to the same objects are called aliases of each other

-this means that one object can be accessed through multiple reference variables

The string class

Since strings are so common, we don’t have to use the new operator to create a String Object

-String title = “Java software solutions”;

This special syntax works only for Strings

Each string literal (in “”) represents a string object

Whats the difference?

Strings literal

String n1 = “Steve Jobs”;

String n2 = “Steve jobs”;

System.out.println (n1==n2); // true

String Object

String n3 = new String (“Steve Jobs”);

String n4 = new String (“Steve Jobs”);

System.out.println (n3 == n4); // false

String Indexes

Sometimes it is helpful to refer to a particular character within a string

To do this specify the characters numeric index

Indexes begin at zero in each string

-in the string “Hello”, the character ‘H’ is at index 0 and the ‘o’ is at index 4

String Objects and Methods

Once a string object has been created neither its value nor its length can be changed

-An object of the String class is said to be immutable

Several methods of the String class return new String objects that are modified versions of the original