Project 4 Assignment:

**1)**

public class String-CharAt {

public static void main(String [] args) {

String s = “hello”;

char c = s.charAt(2);

c++; //char +, +1, ++

s = “he” + c + c + “o”; //String +

}

}

c, s?

c = “m”

s = “hemmo”

**2)**

public class String-Substring {

public static void main(String [] args) {

String t = “hee dee hee dee hee dee hee”;

int x = t.indexOf(“dee”); // indexOf(str)

String u = t.substring(0,x); // substring(x, y)

u = u + “haw”;

x = t.indexOf(“dee”); //second call of indexOf(str)

}

}

u, x?

x = 4

u = “hee haw”

**3)**

public class String-IndexI {

public static void main(String [] args) {

String s = “hi dee hi dee hi”; // find four blank space first!

String t = “”;

int x = s.indexOf(“hi dee”, 0); // indexOf(str, x)

while(x>=0) {

t = t + “ho dee”;

x = s.indexOf(“hi dee”, x+1); // indexOf(str, x)

}

t = t + “ho”;

}

}

t,x?

t = “ho deeho deeho

x = -1

**4)**

public class String-Equals {

public static void main(String [] args) {

String s = “hibbity” +

“hibbity”;

int i=0;

int count = 0;

while(i<s.length()-3) { // length()

if(s.substring(i,i+3).equals(“bit”)) { //substring(x,y), equals(str)

count++;

}

i++;

}

}

}

count, i?

count = 2

i = 11

**5)**

public class String-Assignments {

public static void main(String [] args) {

String s;

String t = null;

String u = "you";

String v = new String("me"); // compare u and v!

String w = u + v;

}

}

s, t, u, v, w?

s is unassigned

t = null

u = “you”

v = “me”

w = “youme”

**6)**

public class String-Commands {

public static void main(String [] args) {

String s = "Call me Ishmael!";

int len = s.length(); //length()

int ishPos = s.indexOf("Ish"); //indexOf(str)

int jackPos = s.indexOf("Jack"); // another indexOf(str)

String ishSub = s.substring(ishPos); //substring(x)!!!

char c = s.charAt(ishPos); // charAt(x)

}

}

ishPos, jackPos, ishSub, c, len?

ishPos = 8

jackPos = -1

ishSub = “Ishmael!”

c = “I”

len = 16

**7)**

// Here is an example that removes a portion of a String,

// and inserts a replacement

public class String-Insert-Delete {

public static void main(String [] args) {

String s = "It was a cold day!";

int start = s.indexOf("cold"); //indexOf(str)

int end = start + "cold".length(); //length()

s = s.substring(0, start) //substring(x,y)

+ "hot"

+ s.substring(end); //substring(x)!

}

}

s, start, end?

s = “It was a hot day!”

start = 9

end = 13

**8)**

// Here is a typical example of a loop used to

// process a String.

// In this example, the loop visits each character

// in the String once.

public class String-Processing {

public static void main(String [] args) {

String s = "Call me Ishmael!";

int aCount = 0, i = 0;

char c = 0;

for( ; i<s.length(); i++) { //length()

c = s.charAt(i); //charAt(int)

if(c == 'a') {

aCount++;

}

}

}

}

aCount, c?

aCount = 2

c = “!”

**9)**

// Here is an example that repeatedly loops through the String,

// processing one word at a time.

public class String-Processing2 {

public static void main(String [] args) {

String s = "Ships at a distance have every man's wish.";

int spacePos1 = 0;

int spacePos2 = s.indexOf(" "); //indexOf(“<space>”)

String hyphenated = ""; //empty string

while(spacePos2>=0) {

String word = s.substring(spacePos1,spacePos2); //substring(x, y)

hyphenated = hyphenated + word + "-";

spacePos1 = spacePos2 + 1;

spacePos2 = s.indexOf(" ", spacePos1); //indexOf(str, x)

}

if(spacePos1<s.length()) {

hyphenated = hyphenated + s.substring(spacePos1); //substring(x)!

}

}

}

spacePos1, spacePos2, hyphenated?

spacePos1 = 37

spacePos2 = -1

hyphenated = “Ships-at-a-distance-have-every-man’s-wish.”