Stack.java

import java.util.Stack;

public class javaStack {

private static Stack st;

javaStack(){

st = new Stack();

}

public boolean IsEmpty(){

if(st.isEmpty()){

return true;

}return false;

}

public void Push(String str){

st.push(str);

}

public String Pop(){

if(!st.isEmpty()){

return (String)st.pop();

}return null;

}

public static void main(String[] args) {

javaStack js = new javaStack();

js.Push("test");

js.Push("This is a ");

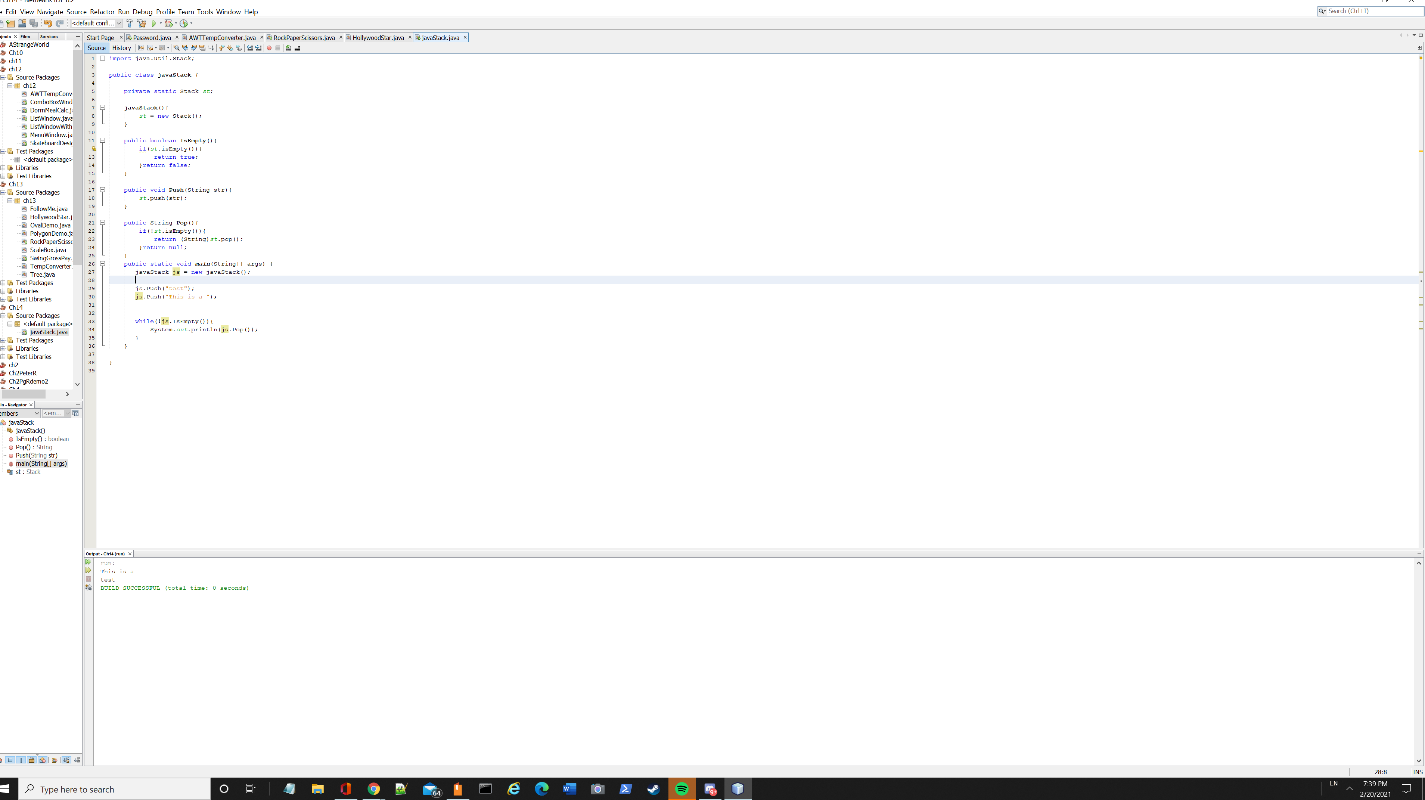
while(!js.IsEmpty()){

System.out.println(js.Pop());

}

}

}



Queue.java

import java.util.\*;

public class javaQueue {

public static void main(String[] args) {

Queue<String> invite = new LinkedList();

Queue<String> noInvite = new LinkedList();

invite.add("Mark");

invite.add("Betty");

invite.add("Joseph");

noInvite.add("Skylar");

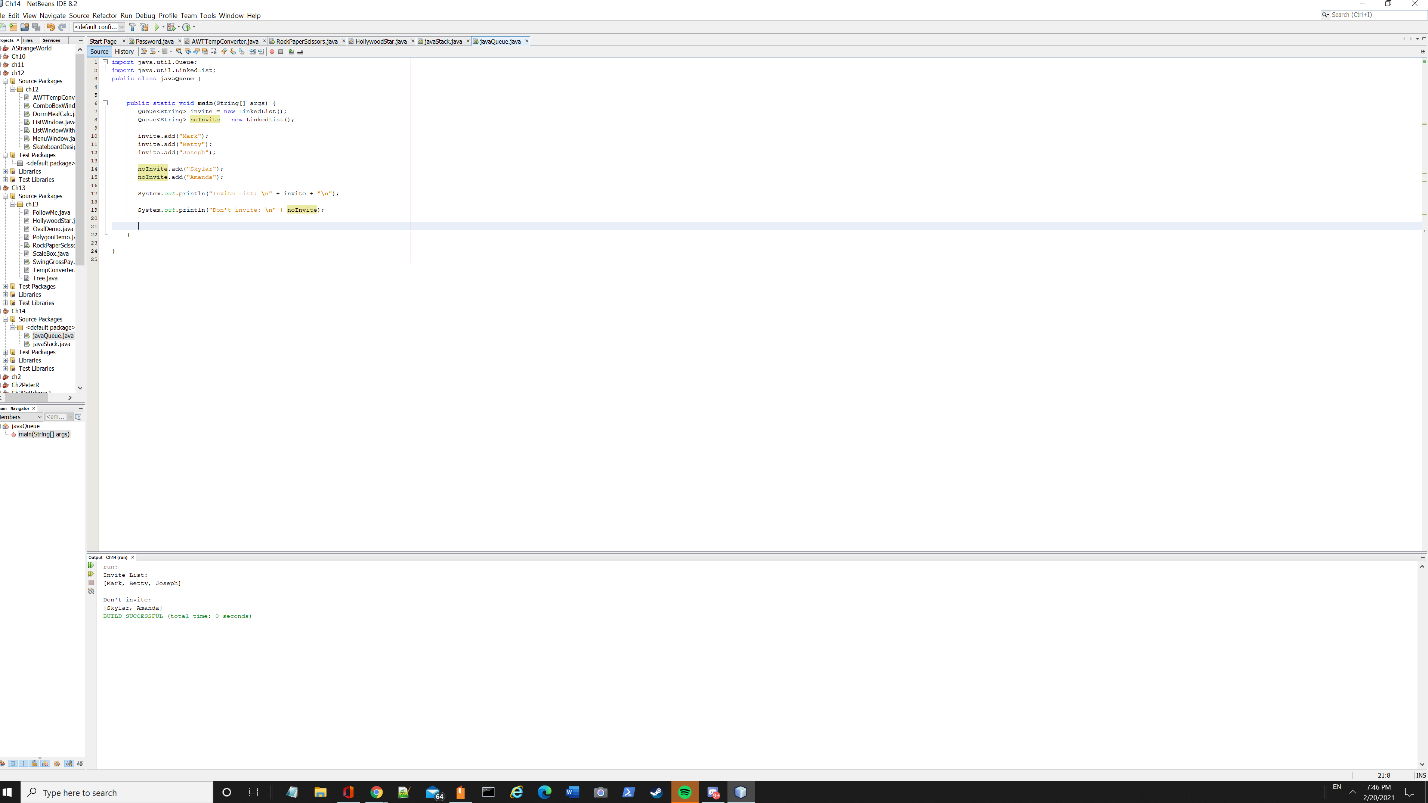
noInvite.add("Amanda");

System.out.println("Invite List: \n" + invite + "\n");

System.out.println("Don't invite: \n" + noInvite);

}

}



Challenge #6

import java.util.\*;

public class characterCounter {

public static void charCount(char a, String[] b){

int count = 0;

String s = "";

for (String n:b){

s += n;

}

char[] c = s.toCharArray();

for(int i = 0; i < c.length; i++){

if(c[i] == a){

count++;

}

}

System.out.println(count);

}

public static void main(String[] args) {

Scanner scanner = new Scanner(System.in);

String[] j = new String[3];

j[0] = "ab";

j[1] = "adc";

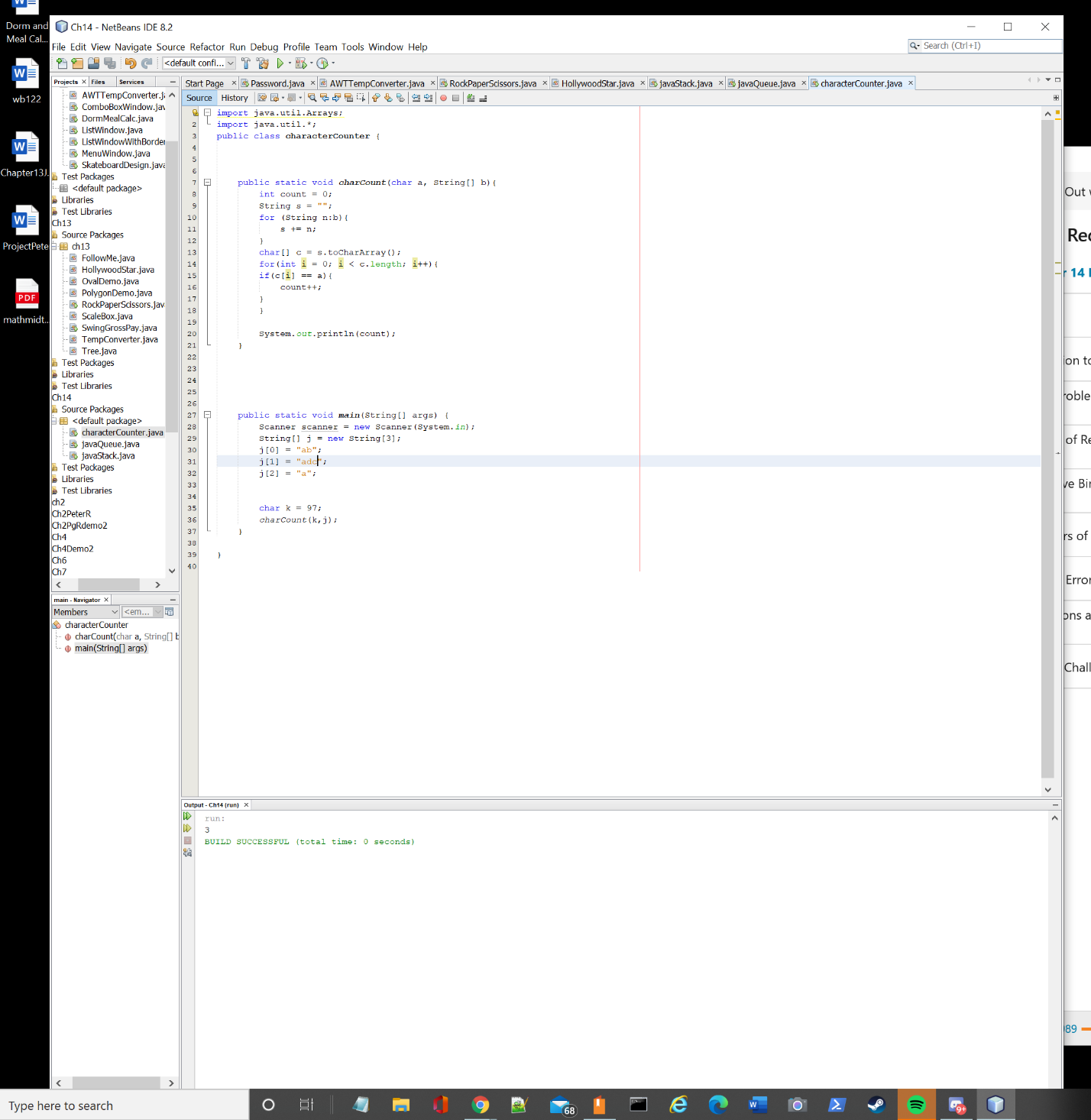
j[2] = "a";

char k = 97;

charCount(k,j);

}

}



Challenge #8

public class Summation {

public static void Summation(int a){

int sum = 0;

int count = 0;

for (int b = a; b > 0; count++){

sum = sum + count;

System.out.println("Adding number " + count + " to sum = " + sum);

if(count == a){

System.out.println("Complete");

System.exit(0);

}

}

}

public static void main(String[] args) {

int c = 50;

Summation(c);

}

}

