Assignment#2

Zehua Li

**#1.**

public class Dsb {

public Integer find(String s,Character c){

Integer res = 0;

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

if(c.equals(s.charAt(i))){

res += 1;

}

}

return res;

}

public static void main(String[] args) {

Dsb dsb = new Dsb();

System.out.println(dsb.find("North",'r'));

}

}

#2.

public class Qtwo {

public String find(int[] a) {

int max = Integer.MIN\_VALUE;

int min = Integer.MAX\_VALUE;

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

if (a[i] >= max) {

max = a[i];

}

if (a[i] < min) {

min = a[i];

}

}

return "max is "+max+" min is "+min;

}

public static void main(String[] args) {

Qtwo qtwo = new Qtwo();

int[] x = {1,4,43,6};

System.out.println(qtwo.find(x));

}

}

#3.

public class Qthree {

public static int add(int a){

int sum = 0;

if (a < 10){

return a;

}

while (a>0){

sum += a%10;

a = a/10;

}

return add(sum);

}

public static void main(String[] args) {

int a = 985;

int b = add(a);

System.out.println(b);

}

}

#4

1)

public class Assignment2 {

public static double socialSecurityTax(double a){

if(a>8900){

a = a\*0.0062;

}

else{

a = 106800\*0.0062;

}

return a;

}

public static void main(String[] args) {

double a = 99909;

System.out.println(socialSecurityTax(a));

}

}

2)

public class Assignment2b {

public static double insuranceCoverage(int employee) {

//write your code here

if(employee < 35){

return(3);

}

if(employee >=35 && employee <= 50){

return(4);

}

if(employee>50 && employee < 60){

return(5);

}

else{

return(6);

}

}

public static void main(String[] args) {

int employee = 34;

System.out.println(insuranceCoverage(employee)+"%");

}

}

3)

import java.util.HashMap;

public class Assignment2c {

public void sort(int e1, int e2, int e3){

int low = 0;

int median = 0;

int high = 0;

HashMap<Integer,String> employee = new HashMap<>();

employee.put(e1,"John");

employee.put(e2,"Mike");

employee.put(e3,"Yang");

if(e1-e2>0&&e3-e2<0){

median = e2;

high = e1;

low = e3;

//System.out.println(low+median+high);

}

else if(e1-e2<0&&e3-e2>0){

median = e2;

high = e3;

low = e1;

//System.out.println(low+median+high);

}

else if(e2-e1<0&&e3-e1>0){

median = e1;

high = e3;

low = e2;

//System.out.println(low+median+high);

}

else if(e3-e1<0&&e2-e1>0){

median = e1;

high = e2;

low = e3;

//System.out.println(low+median+high);

}

else if(e1-e3<0&&e2-e3>0){

median = e3;

high = e2;

low = e1;

//System.out.println(low+median+high);

}

else if(e2-e3<0&&e1-e3>0){

median = e3;

high = e1;

low = e2;

//System.out.println(low+median+high);

}

System.out.println(employee.get(low)+" "+employee.get(median) + " " + employee.get(high));

}

public static void main(String[] args) {

int e1 = 3;

int e2 = 4;

int e3 = 9;

Assignment2c assignment2c = new Assignment2c();

assignment2c.sort(e1,e2,e3);

}

}

Extra credit:

I think in Java, we cannot directly copy the value, because like what I learned from the class, the objects are about references, if we let some object equals to some value, that only means that this object is point to that value, like the object is pointing to a pool. So that the function in this code, even if we seem that we have swap the value of the ‘x’ and ‘y’, we did not swap their reference so that their value does not come with them.