תכנות מערכות בשפת Java

תרגיל מספר 2 פתרונות דוד אלקובי

1)

public static int sumDigits(int num1) {  
 int sum = 0;  
 while (num1 > 0) {  
 sum = sum + num1 % 10;  
 num1 /= 10;  
 }  
 return sum;  
}  
  
public static void main(String[] args) {  
 int left, right, sum = 0, max = 0, num\_max = 0;  
 Scanner in = new Scanner(System.*in*);  
 System.*out*.println("enter a number:");  
 left = in.nextInt();  
 System.*out*.println("enter a bigger number:");  
 right = in.nextInt();  
 for (int i = left; i <= right; i++) {  
 int current\_sum = *sumDigits*(i);  
 if (current\_sum > max) {  
 max = current\_sum;  
 num\_max = i;  
 }  
  
  
 }  
 System.*out*.println(num\_max);

2)

public static void digits (int num) {  
 int digit = 0, cntEven = 0, cntOdd = 0;  
 while (num > 0) {  
 digit = num % 10;  
 if (digit % 2 == 0)  
 cntEven++;  
 else  
 cntOdd++;  
 num /= 10;  
 }  
 if (cntEven>cntOdd)  
 System.*out*.println("גדול");  
 else if (cntOdd>cntEven)  
 System.*out*.println("קטן");  
 else System.*out*.println("שווה");  
  
}  
  
 public static void main(String[] args) {  
 *digits*(713);

3)

int reversed=0;  
 while (x>0){  
 reversed=reversed\*10+x%10;  
  
 x /= 10;  
 }  
 System.*out*.println(reversed);  
 return reversed;  
  
  
}  
  
public static void main(String[] args) {  
 *reverse*(3);

4)

public static long merge (int a,int b) {  
 long result = 0;  
 long reverseA = *reverse*(a);  
 long reverseB = *reverse*(b);  
 while (reverseA > 0 || reverseB > 0) {  
 if (reverseA>0){  
 result = result \* 10 + reverseA % 10;  
 reverseA/=10;}  
 if (reverseB>0)  
 result = result \* 10 + reverseB % 10;  
 reverseB/=10;  
 }  
 System.*out*.println(result);  
 return result;

5)

6)

int left\_two\_digits, right\_two\_digits;  
for (int i = 1000; i <=9900; i++) {  
 left\_two\_digits=i/100;  
 right\_two\_digits=i%100;  
 if (Math.*pow*(left\_two\_digits+right\_two\_digits,2)==i)  
 System.*out*.println(i);

7)

double perimeter\_r1,perimeter\_R1,ringArea,ringArea2;  
int r1,R1;  
Scanner in=new Scanner(System.*in*);  
r1=in.nextInt();  
R1=in.nextInt();  
ringArea= (Math.*PI*\*Math.*pow*(R1,2))-(Math.*PI*\*Math.*pow*(r1,2));  
perimeter\_R1= 2\*Math.*PI*\*R1;  
perimeter\_r1= 2\*Math.*PI*\*r1;  
ringArea2=((perimeter\_R1+perimeter\_r1)/2)\*(R1-r1);  
System.*out*.println("area 1: "+ringArea);  
System.*out*.println("area 2: "+ringArea2);

8)

public static double CtoF(double C) {  
 double F;  
 F = ((9 \* C) / 5) + 32;  
 return F;  
}  
  
static public double FtoC(double F) {  
 double C;  
 Scanner in = new Scanner(System.*in*);  
 F = in.nextDouble();  
 C = ((F - 32) \* 5) / 9;  
 return C;  
}  
  
public static void main(String[] args) {  
 for (int i = -273; i <= 101; i += 11) {  
 System.*out*.println(i +" "+'\t'+*CtoF*(i) );

9)

public static void main(String[] args) {  
 for (int i = 100; i <1000 ; i++) {  
 int hundreds=i/100,tens=(i%100)/10,units=i%10;  
 if (Math.*pow*(hundreds,3)+Math.*pow*(tens,3)+Math.*pow*(units,3)==i)  
 System.*out*.println(i);

10)

public static long reversed (int x) {  
 int reversed = 0;  
 while (x > 0) {  
 reversed = reversed \* 10 + x % 10;  
  
 x /= 10;  
 }  
 return reversed;  
}  
  
public static void main(String[] args) {  
 for (int i = 10000; i <=99999 ; i++) {  
 if (i\*4==*reversed*(i))  
 System.*out*.println(i);

11)

int ID;  
Scanner in=new Scanner(System.*in*);  
ID=in.nextInt();  
for (int i = 1; i <=8 ; i++) {  
 int bikoret=ID%10;  
 int first=(ID%100)/10;  
 int second=(ID%1000)/100;  
 int third=(ID%10000)/1000;  
 int forth=(ID%100000)/10000;  
 int fifth=(ID%1000000)/100000;  
 int sixth=(ID%10000000)/1000000;  
 int seven=(ID%100000000)/10000000;  
 int eight=(ID%1000000000)/100000000;  
 int check=(first\*2)+second+(third\*2)+forth+(fifth\*2)+sixth+(seven\*2)+eight;  
 if ((check<10)&&10-check==bikoret){  
 System.*out*.println("valid ID");}  
  
 if (check>9&&(100-check)%10==bikoret)  
 System.*out*.println("valid ID");