

函式

請將 1.程式執行結果截圖置入作業中、2.程式原始檔置入作業中

一、請參考投影片內容，編寫下列程式

- 1) 利用遞迴的方式設計一個函式 `double myPower(double x,int y)`，計算出 `x` 的 `y` 次方。(若 `y<0`，則回傳 0)

```
import java.lang.*;
import java.io.*;
public class test_01
{
    public static void main(String args[])
    {
        int x,y;
        double Ans;
        Console console=System.console();

        System.out.print("計算x的y次方\n請輸入x=");
        x=Integer.parseInt(console.readLine());
        System.out.print("請輸入y=");
        y=Integer.parseInt(console.readLine());
        Ans=power(x,y);
        System.out.println(+x+"的" + y + "次方=" + Ans);
    }
    public static double power(double x,int y)
    {
        double a;
        int i;

        if(y==0)
            return 1;
        else if(y==1)
            return x;
        else
            return (x*power(x,y-1));
    }
}
```

```
C:\windows\system32\CMD.exe
Microsoft Windows [版本 10.0.17134.1069]
(c) 2018 Microsoft Corporation. 著作權所有，並保留一切權利。
C:\Users\penit>cd /test
C:\test>javac test_01.java
C:\test>java test_01
計算x的y次方
請輸入x=3
請輸入y=4
3的4次方=81.0
C:\test>
```

- 2) 設計一個函式 `void print_total_mul(int a,int b)`，列印 `ab` 乘法表，例如 `a=5`、`b=8` 時，輸出如下（當 `a=8`、`b=5` 時，輸出也相同）：（當 `a=1`、`b=9` 時，恰恰為九九乘法表）

【執行結果】

5*5=25	5*6=30	5*7=35	5*8=40
6*5=30	6*6=36	6*7=42	6*8=48
7*5=35	7*6=42	7*7=49	7*8=56
8*5=40	8*6=48	8*7=56	8*8=64

```

import java.lang.*;
import java.io.*;

public class test_02
{
    public static void main(String args[])
    {
        int a,b;

        Console console=System.console();

        a=Integer.parseInt(console.readLine());

        b=Integer.parseInt(console.readLine());

        print_total_mul(a,b);

    }
    public static void print_total_mul(int a,int b)
    {
        int i,j;
        if(b>=a)
            for(i=a;i<=b;i++)
            {
                for(j=a;j<=b;j++)
                {
                    System.out.print(+i+"*"+j+"="+i*j+"\t");
                }
                System.out.println("\n");
            }
        else
            for(i=b;i<=a;i++)
            {
                for(j=b;j<=a;j++)
                {
                    System.out.print(+i+"*"+j+"="+i*j+"\t");
                }
                System.out.println("\n");
            }
    }
}

```

```

C:\windows\system32\CMD.exe

C:\Users\penit>cd /test
C:\test>javac test_02.java
C:\test>java test_02
5
8
5*5=25  5*6=30  5*7=35  5*8=40
6*5=30  6*6=36  6*7=42  6*8=48
7*5=35  7*6=42  7*7=49  7*8=56
8*5=40  8*6=48  8*7=56  8*8=64

C:\test>java test_02
8
5
5*5=25  5*6=30  5*7=35  5*8=40
6*5=30  6*6=36  6*7=42  6*8=48
7*5=35  7*6=42  7*7=49  7*8=56
8*5=40  8*6=48  8*7=56  8*8=64

C:\test>

```

- 3) 使用多載技術，設計兩個函式 `double maxEle(int x[][])` 與 `double maxEle(double x[][])`，可找出二維陣列之最大值並回傳。

```
import java.lang.*;
import java.io.*;
import java.lang.Math;
public class test_03
{
    public static void main(String args[])
    {
        int m1[][]=new int[3][3];
        double m2[][]=new double[3][3];
        int i,j,ans1;

        Console console=System.console();

        for(i=0;i<3;i++)
        {
            for(j=0;j<3;j++)
            {
                m1[i][j]=(int) ((Math.random()*49)+1);
                m2[i][j]=(int) ((Math.random()*49)+1);
            }
        }
        System.out.println("初始亂數\n");
        for(i=0;i<3;i++)
        {
            for(j=0;j<3;j++)
            {
                System.out.print(m1[i][j]+" ");
            }
        }
        System.out.println("\n\n");
        for(i=0;i<3;i++)
        {
            for(j=0;j<3;j++)
            {
                System.out.print(m2[i][j]+" ");
            }
        }
        System.out.println("\n\n");
        System.out.println("\nm1最大值:"+maxEle(m1));
        System.out.println("\nm2最大值:"+maxEle(m2));
    }
    public static int maxEle(int[][] m)
    {
        int i,j,max=-1;

        for(i=0;i<3;i++)
        {
            for(j=0;j<3;j++)
            {
                if(m[i][j]>max)
                    max = m[i][j];
            }
        }
        return max;
    }
    public static double maxEle(double[][] m)
    {
        int i,j;
        double max=-1;
        for(i=0;i<3;i++)
        {
            for(j=0;j<3;j++)
            {
                if(m[i][j]>max)
                    max = m[i][j];
            }
        }
    }
}
```

C:\windows\system32\CMD.exe

C:\test>java test_03

初始亂數

40 27 45 26 10 43 4 15 43

12.0 17.0 15.0 5.0 26.0 23.0 2.0 8.0 13.0

m1最大值:45

m2最大值:26.0

C:\test>