

### 3. สร้างโปรแกรม การคอมไพล์ (compile) และการรัน (run) โปรแกรมจาวา

#### 1) สร้างโปรแกรมจาวา โดยใช้โปรแกรม text editor ชื่อ EditPlus

สร้างไฟล์โปรแกรมใหม่ เลือก File> new โดยเลือกประเภท java

โปรแกรม Editplus จะเตรียมคำสั่งให้บางส่วน ให้ปรับแก้โปรแกรมเป็นดังนี้

```
class hello {
    public static void main(String[] args) {
        System.out.print("Hello World!");
        System.out.println(" I am JAVA.");
    }
}
```

#### 2) บันทึกโปรแกรมลงไฟล์

ในขั้นต้น Editplus ตั้งชื่อไฟล์เป็น Noname1.java

ให้เปลี่ยนชื่อไฟล์เป็น hello โดยใช้ File>Save as ตามด้วยชื่อ hello.java (ชื่อไฟล์เป็นชื่อเดียวกับชื่อคลาส)

อย่าลืม เปลี่ยนไฟล์เดอริไปในพื้นที่ของเราคือ d:\Java\_me

หมายเหตุ บันทึกโปรแกรมที่แก้แล้ว ใช้ File>Save

นำโปรแกรมเดิม ขึ้นมาแก้ไข หรือรัน ใช้ File>Open

#### 3) คอมไพล์ รันโปรแกรม และแสดงผลลัพธ์

	1	2	3
1	Helio	World!	I am JAVA.
2			
3			

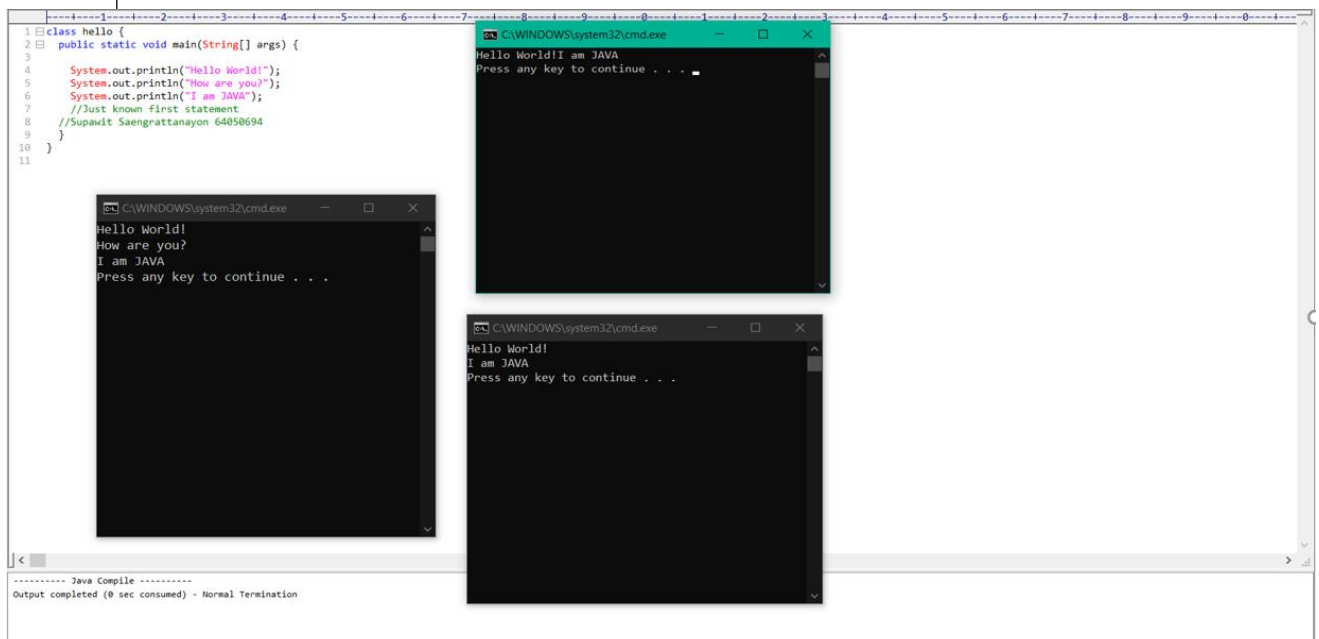
#### 4) เปลี่ยนคำสั่ง print เป็น println (บรรทัด Hello World!) คอมไพล์ รัน และแสดงผลลัพธ์

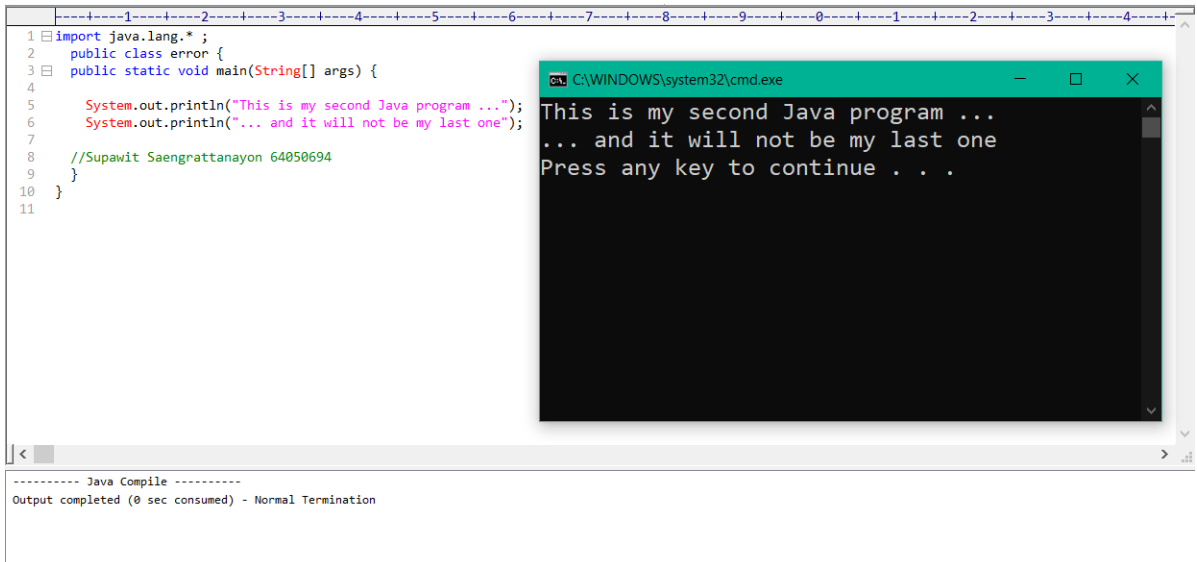
	1	2	3
1	Helio	World!	
2	I	am	JAVA.
3			

#### 5) เพิ่มคำสั่งใหม่ ให้แสดงข้อความ "How are you?" ในบรรทัดที่ถัดจาก "Hello, World!" แล้วคอมไพล์และรัน

	1	2	3
1	Helio	World!	
2	How	are	you?
3	I	am	JAVA.
4			
5			

#### 6) เพิ่มคอมเมนต์ใหม่ข้อความ "Just known first statement" ในขอบเขตเมทอด main แล้วคอมไพล์และรัน โดยผลการทำงานยังถูกต้องเหมือนข้อ 5)





Lab 1: ชื่อ หวน กฤษณ์ แสงรัตนานนท์ รหัสประจำตัว 64050694 กลุ่ม 6 หน้า 4

รายการทดลองแก้ไขคำสั่ง	คอมไพล์: ผ่าน ✓ ไม่ผ่าน ✗	รัน: รันได้ ✓ รันไม่ได้ ✗	ข้อความที่แจ้ง ตรงกับที่ควรจะเป็นหรือไม่ ถ้ารันได้ ผลลัพธ์ตรงกับที่คาดหวังหรือไม่
1) ลบเครื่องหมายวงเล็บปีกกาเปิด { ออกหนึ่งอัน (ทดลองที่ละอัน) - { อันแรก - { อันที่สอง	✗	✗	
2) ลบเครื่องหมายวงเล็บปีกกาปิด } ออกหนึ่งอัน (ทดลองที่ละอัน) - } อันแรก - } อันที่สอง	✗	✗	
3) แก้คำว่า main เป็น mian (เป็นชื่อที่ไม่มีอยู่จริง)	✓	✗	
4) ลบคำว่า static	✓	✗	
5) ลบคำว่า public public คำแรก public คำที่สอง	✓ ✓	✓ ✗	
6) ลบคำว่า System คำแรก	✗	✗	
7) เว้นวรรค ในคำสั่ง println ( ) แรก หน้าจุด หลัง System หน้าจุด หลัง out เช่น System .out .println("Hi");	✓	✓	
8) เว้นวรรค ในคำสั่ง println ( ) แรก หลังจุด หลัง System หลังจุด หลัง out เช่น System.out .println("Hi");	✓	✓	
9) เปลี่ยน println เป็น println (เป็นชื่อที่ไม่มีอยู่จริง)	✗	✗	
10) เปลี่ยน println เป็น print คำสั่ง System.out.print ถูกต้อง แต่ผลลัพธ์ อาจจะตรงหรือไม่ตรงกับที่คาดหวัง	✓	✓	เป็นเทคนิคในการตรวจสอบผลการทำงาน โดย เพิ่มคำสั่งให้พิมพ์อะไรบางอย่างให้รู้ ว่าทำงาน ถึงจุดไหนของโปรแกรม
	คอมไพล์:	รัน:	ข้อความที่แจ้ง

รายการทดลองแก้ไขคำสั่ง	ผ่าน ✓ ไม่ผ่าน ✗	รันได้ ✓ รันไม่ได้ ✗	ตรงกับที่ควรจะเป็นหรือไม่ ถ้ารันได้ ผลลัพธ์ตรงกับที่คาดหวังหรือไม่
11) เพิ่ม ; ; หลังคำสั่ง println() ที่ 2	✓	✓	
12) เพิ่มช่องว่าง 5 ช่อง หลัง main ก่อน (	✓	✓	
13) ลบวงเล็บ ( หลัง main ออก	✗	✗	
14) เพิ่ม ) หลัง args	✗	✗	
15) แก่ชื่อ args เป็น arg	✓	✓	
16) เปลี่ยน import java.lang.* ; เป็น import java.* ;	✓	✓	
17) เปลี่ยน import java.lang.* ; เป็น import java.lang.point.* ;	✗	✗	

5) เขียน จัดวางคำสั่งในโปรแกรมใหม่ ให้ ง่ายต่อการอ่าน และ ง่ายต่อการทำความเข้าใจ

```
class mathIso
{
    public static void main ( String[] args )
    {
        System.out.println("I study Applied Mathematics");
        System.out.println("and study computer languages.");
        System.out.println("I will be expert in mathematics");
        System.out.println("and I will be the programmer.");
    }
}
```

เขียนจัดวางใหม่ดังนี้

```
class mathIso {
    public static void main(String[] args){
        System.out.println("I study Applied Mathematics");
        System.out.println("and study computer languages.");
        System.out.println("I will be expert in mathematics");
        System.out.println("and I will be the programmer.");
    }
}
```