

ใบงานการทดลองที่ 2

เรื่อง ภาษาจาวากับการโปรแกรมเชิงวัตถุ

1. จุดประสงค์ทั่วไป

- 1.1. รู้และเข้าใจการโปรแกรมเชิงวัตถุด้วยคำสั่งพื้นฐานในภาษาจาวา
- 1.2. รู้และเข้าใจเพื่อศึกษาและทดลองสร้างโปรแกรมเชิงวัตถุโดยใช้ภาษาการโปรแกรมเชิงวัตถุใหม่ๆ

2. เครื่องมือและอุปกรณ์

เครื่องคอมพิวเตอร์ 1 เครื่อง ที่ติดตั้งโปรแกรม Eclipse

3. ทฤษฎีการทดลอง

- 3.1. คำสั่งที่ใช้ในการแสดงผลบนแถบ Console คือคำสั่งใด

.....

.....

- 3.2. คำสั่งการคอมเมนต์แบบบรรทัดเดียว และแบบหลายบรรทัด คืออะไร ?

- คำสั่งคอมเมนต์แบบบรรทัดเดียว คือ //

- คำสั่งคอมเมนต์แบบหลายบรรทัด คือ /*

.....

- 3.3. ประเภทข้อมูลแบบ Integer คืออะไร? อธิบายพร้อมยกตัวอย่างประกอบเพื่อแสดงค่าภายในตัวแปร

คือประเภทข้อมูลชนิดจำนวนเต็มมีขนาดข้อมูล 32 bit (4 byte) ตัวอย่าง int number = 5;

System.out.println("Your number is" + number);

.....

- 3.4. ประเภทข้อมูลแบบ Double คืออะไร? อธิบายพร้อมยกตัวอย่างประกอบเพื่อแสดงค่าภายในตัวแปร

คือประเภทข้อมูลตัวเลขรูปแบบทศนิยมมีขนาดข้อมูล 64 bit (8 byte) ตัวอย่าง double sum = 1425.55;

System.out.println("Your sum is" + sum);

.....

- 3.5. ประเภทข้อมูลแบบ Float คืออะไร? อธิบายพร้อมยกตัวอย่างประกอบเพื่อแสดงค่าภายในตัวแปร

คือประเภทข้อมูลตัวเลขรูปแบบทศนิยมมีขนาดข้อมูล 32 bit (4 byte) ตัวอย่าง float sum = 555.56;

System.out.println("Your sum is" + sum);

.....

- 3.6. ประเภทข้อมูลแบบ Bool คืออะไร? อธิบายพร้อมยกตัวอย่างประกอบเพื่อแสดงค่าภายในตัวแปร

คือประเภทข้อมูลที่ใช้เปรียบเทียบข้อมูล true และ false มีขนาดข้อมูล 1 bit ตัวอย่าง Boolean t = true; Boolean f = false;

System.out.println("Your mother is female" + t);

.....

- 3.7. ประเภทข้อมูลแบบ Char คืออะไร? อธิบายพร้อมยกตัวอย่างประกอบเพื่อแสดงค่าภายในตัวแปร

คือประเภทข้อมูลที่เก็บข้อมูลเป็นชนิดอักขระได้แก่ตัวอักษร, ตัวเลข และสัญลักษณ์ต่างๆ ตัวอย่าง char female = 'f';

สามารถเก็บข้อมูลได้เพียง 1 ตัวอักษรเท่านั้นมีขนาดข้อมูล 16 bit (2 byte)

System.out.println("The female initials are" + female);

.....

- 3.8. ประเภทข้อมูลแบบ String คืออะไร? อธิบายพร้อมยกตัวอย่างประกอบเพื่อแสดงค่าภายในตัวแปร

คือประเภทข้อมูลที่เก็บข้อมูลเป็นชนิดอักขระได้แก่ตัวอักษร, ตัวเลข และสัญลักษณ์ต่างๆ ตัวอย่าง String message = "Welcome";

สามารถเก็บข้อมูลได้เพียง 1 ตัวอักษร

System.out.println("Hi," + message);

.....

- 3.9. Widening Casting คืออะไร? อธิบายพร้อมยกตัวอย่างประกอบ

เป็นการกำหนดค่าของ Variable ที่มี Data Type ลำดับต่ำกว่าให้กับ Variable ที่สูงกว่า ตัวอย่าง byte < short < int < long < float < double

.....

- 3.10. Narrowing Casting คืออะไร? อธิบายพร้อมยกตัวอย่างประกอบ

เป็นการกำหนดค่าของ Variable ที่มี Data Type ลำดับสูงกว่าให้กับ Variable ที่ต่ำกว่า

ตัวอย่าง double < float < long < int < short < byte

3.11. จงอธิบายความหมายและยกตัวอย่างการใช้งานตัวดำเนินการทางคณิตศาสตร์ดังต่อไปนี้

ตัวดำเนินการ	ความหมาย	ตัวอย่างการใช้งาน
+	เพิ่มค่า	$x = 1 + 2$
-	ลบค่า	$x = 20 - 10$
*	คูณค่า	$x = 8 * 8$
/	หารค่า	$x = 10 / 2$
++	การเพิ่มค่าของจำนวนเต็ม 1	$x = x++;$
--	การลดค่าของจำนวนเต็ม 1	$x = x--;$
%	การหารแล้วเอาแต่เศษ	$x = 11 \% 3$

3.12. จงอธิบายความหมายและยกตัวอย่างการใช้งานตัวดำเนินการเพื่อกำหนดค่าดังต่อไปนี้

ตัวดำเนินการ	ความหมาย	ตัวอย่างการใช้งาน
=	เป็นการกำหนดค่าให้กับตัวแปร	$x = 5$
+=	นำตัวแปรด้านซ้ายมาบวกเพิ่มก่อนที่จะกำหนดค่าให้	$a += 10$ มีค่าเท่ากับ $a = a + 5$
-=	นำตัวแปรด้านซ้ายมาลบออกก่อนที่จะกำหนดค่าให้	$a -= 10$ มีค่าเท่ากับ $a = a - 5$
*=	นำตัวแปรด้านซ้ายมาคูณก่อนที่จะกำหนดค่าให้	$a *= 10$ มีค่าเท่ากับ $a = a * 5$
/=	นำตัวแปรด้านซ้ายมาหารก่อนที่จะกำหนดค่าให้	$a /= 10$ มีค่าเท่ากับ $a = a / 5$
%=	นำตัวแปรด้านซ้ายมาหารเก็บเศษก่อนที่จะกำหนดค่าให้	$a \% = 10$ มีค่าเท่ากับ $a = a \% 5$
&=	นำตัวแปรด้านซ้ายมาทำการ AND ระดับบิตก่อนที่จะกำหนดค่าให้	$a \& = 10$ มีค่าเท่ากับ $a = a \& 5$
=	นำตัวแปรด้านซ้ายมาทำการ or ระดับบิตก่อนที่จะกำหนดค่าให้	$a = 10$ มีค่าเท่ากับ $a = a 5$

3.13. จงอธิบายความหมายและยกตัวอย่างการใช้งานตัวดำเนินการเพื่อเปรียบเทียบดังต่อไปนี้

ตัวดำเนินการ	ความหมาย	ตัวอย่างการใช้งาน
==	การตรวจสอบค่าทั้งสองว่ามีค่าเหมือนกันไหมถ้าเท่ากันจะได้ค่า true	$(1 == 1)$
>	การตรวจสอบค่าทั้งสองว่าค่าทางด้านซ้ายมีค่ามากกว่าทางด้านขวาหรือไม่ถ้าใช่จะได้ค่า true	$(2 != 3)$
>=	การตรวจสอบค่าทั้งสองว่าค่าทางด้านซ้ายมีค่ามากกว่าหรือเท่ากับทางด้านขวาหรือไม่ถ้าใช่จะได้ค่า true	$(5 > 1)$
<	การตรวจสอบค่าทั้งสองว่าค่าทางด้านซ้ายมีค่ามากกว่าทางด้านขวาหรือไม่ถ้าใช่จะได้ค่า true	$(1 < 5)$
<=	การตรวจสอบค่าทั้งสองว่าค่าทางด้านซ้ายมีค่าน้อยกว่าหรือเท่ากับทางด้านขวาหรือไม่ถ้าใช่จะได้ค่า true	$(5 >= 5)$
!=	การตรวจสอบค่าทั้งสองว่ามีค่าไม่เท่ากันไหมถ้าไม่เท่ากันจะได้ค่า true	$(5 <= 5)$

3.14. จงอธิบายความหมายและยกตัวอย่างการใช้งานตัวดำเนินการตรรกศาสตร์ดังต่อไปนี้

ตัวดำเนินการ	ความหมาย	ตัวอย่างการใช้งาน
&&	จะเป็นค่า true ก็ต่อเมื่อค่าทั้งสองค่าเป็น true	$(a \&\& b)$ ได้ค่า false
	จะเป็นค่า false ก็ต่อเมื่อค่าทั้งสองค่าเป็น false	$(a b)$ ได้ค่า true
!	จะทำการกลับค่าจาก true เป็น false หรือจาก false เป็น true	$!a$ ได้ค่า false

3.15. ประโยคเงื่อนไข - จงเขียน Syntax การใช้งานคำสั่ง if พร้อมยกตัวอย่างประกอบ

```

if (50 > 1) {
    System.out.println("50 is greater than 1");
}

```

3.16. ประโยคเงื่อนไข - จงเขียน Syntax การใช้งานคำสั่ง if else พร้อมยกตัวอย่างประกอบ

```

if(13%2==0){
    System.out.println("even number");
}
else{
    System.out.println("odd number");
}

```

3.17. ประโยคเงื่อนไข - จงเขียน Syntax การใช้งานคำสั่ง if , else if และ else พร้อมยกตัวอย่างประกอบ

```

if (เงื่อนไขที่1) {
    // โค้ดที่จะดำเนินการหากเงื่อนไขเป็นจริง
} else if (เงื่อนไขที่2) {
    // โค้ดที่จะดำเนินการหากเงื่อนไข 1 เป็นเท็จและเงื่อนไข 2 เป็นจริง
} else {
    // โค้ดที่จะดำเนินการหากเงื่อนไข 1 เป็นเท็จและเงื่อนไข 2 เป็นเท็จ
}

```

3.18. ประโยคเงื่อนไข - จงเขียน Syntax การใช้งานคำสั่ง if แบบ Short hand พร้อมยกตัวอย่างประกอบ

```

int time = 20;
String result = (time < 18) ? "Good day." : "Good evening.";
System.out.println(result);

```

3.19. ประโยคเงื่อนไข - จงเขียน Syntax การใช้งานคำสั่ง switch พร้อมยกตัวอย่างประกอบ

```
switch () {
```

```
}
```

3.20. การวนรอบ - จงเขียน Syntax การใช้งานคำสั่ง for พร้อมยกตัวอย่างประกอบ

```
for(กำหนดค่าเงื่อนไขปรับค่า){
```

```
//โค้ดที่ต้องการดำเนินการ
```

```
}
```

3.21. การวนรอบ - จงเขียน Syntax การใช้งานคำสั่ง while พร้อมยกตัวอย่างประกอบ

```
while (เงื่อนไข) {
```

```
// โค้ดที่จะดำเนินการ
```

```
// ปรับค่าตัวแปร
```

```
}
```

3.22. การวนรอบ - จงเขียน Syntax การใช้งานคำสั่ง do while พร้อมยกตัวอย่างประกอบ

```
do{
```

```
//โค้ดที่จะดำเนินการ
```

```
//ปรับค่าตัวแปร
```

```
}while (เงื่อนไข);
```

3.23. อธิบายการทำงานของคำสั่ง break พร้อมยกตัวอย่างประกอบ

```
for (int i = 0; i < 10; i++) {                               ใช้เพื่อ"ออกจากคำสั่ง" ของคำสั่งswitch
```

```
if (i == 4) {
```

```
break
```

```
}
```

```
System.out.println(i);
```

```
}
```

3.24. อธิบายการทำงานของคำสั่ง continue พร้อมยกตัวอย่างประกอบ

```
for (int i = 0; i < 10; i++) {                               ใช้เพื่อดำเนินการคำสั่งต่อในขณะลูปอยู่
```

```
if (i == 4) {
```

```
continue;
```

```
}
```

```
System.out.println(i);
```

```
}
```

3.25. อธิบายการทำงานของคำสั่ง Enum

เป็นประเภทข้อมูลที่เป็นชุดของตัวเลขจำนวนเต็มมันใช้กำหนดกลุ่มของค่าคงที่ที่ทำให้มีความหมายและเข้าใจได้ง่ายขึ้นในการเขียนโปรแกรม

4. ลำดับขั้นการปฏิบัติการ

4.1. จงแก้โจทย์ปัญหาดังต่อไปนี้

4.1.1. จงเขียนโปรแกรมเพื่อแสดงเลข 1 จนถึง N ; โดยที่ N คือค่าที่รับมาจากผู้ใช้ และแสดงคำตอบบรรทัดละ 5 ตัวเลขเท่านั้น

Test case 1	Test case 2
Please enter your value : 8 ----- 1, 2, 3, 4, 5 6, 7, 8	Please enter your value : 21 ----- 1, 2, 3, 4, 5 6, 7, 8, 9, 10 11, 12, 13, 14, 15 16, 17, 18, 19, 20 21

4.1.2. ผลงานแสดงกระบวนการทำงานและโค้ดโปรแกรม(ที่ตรงตามผลงาน)

ผลงาน	โค้ดโปรแกรม
	<pre> 1 package lab2; 2 import java.util.*; 3 4 public class lab2_1 { 5 private static Scanner sc; 6 public static void main(String[] args) { 7 int n, i; 8 sc = new Scanner(System.in); 9 System.out.print(" Please Enter any Number : "); 10 n = sc.nextInt(); 11 for(i = 1; i <= n; i++) 12 { 13 System.out.print(i); 14 if (i % 5 == 0){ 15 System.out.println(); 16 }else { 17 System.out.print(", "); 18 } 19 } 20 } 21 }</pre>

4.2. จงแก้ไขข้อปัญหาดังต่อไปนี้

4.2.1. จงเขียนโปรแกรมเพื่อแสดงจำนวนเฉพาะที่อยู่ระหว่าง เลข 2 จนถึง N ; โดยที่ N คือค่าที่รับมาจากผู้ใช้

Test case 1	Test case 2
Please enter your value : 8 ----- 2 = Prime Number 3 = Prime Number 4 = Not Prime : Divide by 2 5 = Prime Number 6 = Not Prime : Divide by 2, 3 7 = Prime Number	Please enter your value : 16 ----- 2 = Prime Number 3 = Prime Number 4 = Not Prime : Divide by 2 5 = Prime Number 6 = Not Prime : Divide by 2, 3 7 = Prime Number

8 = Not Prime : Divide by 2, 4	8 = Not Prime : Divide by 2, 4 9 = Not Prime : Divide by 3 10 = Not Prime : Divide by 2, 5 11 = Prime Number 12 = Not Prime : Divide by 2, 3, 4, 6 13 = Prime Number 14 = Not Prime : Divide by 2, 7 15 = Not Prime : Divide by 2, 3, 5 16 = Not Prime : Divide by 2, 4, 8
--------------------------------	--

4.2.2. ผลงานแสดงกระบวนการทำงานและโค้ดโปรแกรม(ที่ตรงตามผลงาน)

ผลงาน	โค้ดโปรแกรม
	<pre> 1 package lab2; 2 import java.util.*; 3 4 public class lab2_2 { 5 private static Scanner sc; 6 public static void main(String[] args) { 7 int i,n,counter, j ,p ; 8 Scanner scanner = new Scanner(System.in); 9 System.out.print("Please enter your value : "); 10 n=scanner.nextInt(); 11 for(j=2;j<=n;j++){ 12 counter=0; 13 for(i=1;i<=j;i++){ 14 if(j%i==0){ 15 counter++; 16 } 17 } 18 if(counter==2) { 19 System.out.print(j+" = Prime Number "); 20 System.out.println(); 21 }else if (counter!=1) { 22 System.out.print(j+" = Not Prime : Divide by "); 23 for(p=2;p<=j;p++) { 24 if(j % p == 0 && j != p) { 25 System.out.print(p); 26 System.out.print(" , "); 27 } 28 } 29 System.out.println(); 30 } 31 } 32 } 33 } </pre>

--	--

5. สรุปผลการปฏิบัติการ

ได้รู้คำสั่งพื้นฐานของภาษาJava ตัวอย่างเช่นตัวดำเนินการทางคณิตศาสตร์, ตัวดำเนินการทางตรรกศาสตร์และการเขียนsyntax ประโยคเงื่อนไข การเขียนโปรแกรมต่างๆเพื่อเข้าใจโครงสร้างและคำสั่งของภาษาJava

6. คำถามท้ายการทดลอง

6.1. คำสั่งต่างๆ ระหว่างภาษา C และภาษา Java มีความเหมือนหรือแตกต่างกันอย่างไร ?

- ภาษาc จะพัฒนาโปรแกรมแบบProcedural ,ไม่รองรับthreading และoverloading function ,รองรับpointers ,รองรับการเข้าถึงข้อมูลผ่านvalue และreference

- ภาษาjava จะพัฒนาโปรแกรมแบบเชิงวัตถุ(OOP) ,รองรับthreading และmethod overloading ,ไม่รองรับpointers ,รองรับการเข้าถึงข้อมูลผ่านvalue

6.2. จงระบุข้อควรระวังในการใช้คำสั่ง Switch Case

ต้องกำหนดbreak ไว้ภายหลังโค้ดเพื่อหยุดการทำงานของSwitch Case

6.3. การเลือกใช้คำสั่ง For เหมาะกับสถานการณ์ใดมากที่สุด

ใช้กับการวนซ้ำในจำนวนรอบที่แน่นอนเนื่องจากสามารถกำหนดค่าเริ่มต้นเงื่อนไขและการเปลี่ยนแปลงค่า

6.4. คำสั่ง Casting ควรถูกนำมาใช้งานในสถานการณ์ใดที่สุด

ใช้ต่อเมื่อเราจะกำหนดค่าของ Variable ที่มีData Type ที่เราไม่ต้องการให้กับ Variable ที่มีData Type ที่เราต้องการ
