

บทที่ 1

หลักการเขียนโปรแกรมเชิงวัตถุ

1.1 รูปแบบการเขียนโปรแกรมคอมพิวเตอร์ (Programming paradigm)

เป็นที่ทราบกันดีอยู่แล้วว่าโปรแกรมประกอบด้วยคำสั่ง (code) และข้อมูล (data) รูปแบบการเขียนโปรแกรมคอมพิวเตอร์ (Programming paradigm) สามารถแบ่งออกเป็น 2 แนวคิดใหญ่ๆ ได้แก่ การเขียนโปรแกรมเชิงกระบวนการ (Procedural Programming) และการเขียนโปรแกรมเชิงวัตถุ (Object-Oriented Programming)

การเขียนโปรแกรมเชิงกระบวนการ คือ ชุดคำสั่งที่สั่งให้คอมพิวเตอร์ทำงานตามลำดับขั้นตอน ซึ่งให้ความสำคัญกับขั้นตอนกระบวนการที่ทำ มีการแบ่งโปรแกรมออกเป็นส่วนๆ (Function or Procedure) เพื่อช่วยจัดระเบียบของคำสั่งในโปรแกรม ทำให้สามารถตรวจสอบลำดับการทำงานได้ง่าย โปรแกรมย่อยสามารถเรียกใช้โปรแกรมย่อยอื่นๆ หรือเรียกใช้ตัวเองได้ ซึ่งแนวคิดแบบ Procedural Programming นั้นจะแยกส่วนที่เป็นข้อมูล (data) ออกจากคำสั่ง (code) ด้วยเหตุนี้ฟังก์ชันที่ใช้จัดการกับข้อมูลจึงอาจถูกวางไว้อย่างกระจัดกระจาย สร้างความลำบากในการติดตามสืบหาว่าข้อมูลถูกเปลี่ยนแปลงโดยฟังก์ชันใด

การเขียนโปรแกรมเชิงวัตถุให้ความสำคัญกับวัตถุ ซึ่งประกอบด้วยข้อมูล (data) และคุณสมบัติ (properties or methods) และผูก data กับ methods เข้าด้วยกัน โดยวัตถุแต่ละวัตถุสามารถนำมาประกอบกันและทำงานรวมกันได้ โดยผ่านการแลกเปลี่ยนข้อมูลเพื่อใช้ในการประมวลผล และวัตถุยังสามารถส่งข้อมูลที่ได้ไปให้วัตถุอื่นๆ ที่เกี่ยวข้องเพื่อใช้ทำงานต่อไป ความแตกต่างระหว่างโปรแกรมเชิงกระบวนการกับโปรแกรมเชิงวัตถุแสดงดังตารางที่ 1.1

ตารางที่ 1.1 ความแตกต่างระหว่างการเขียนโปรแกรมเชิงกระบวนการกับการเขียนโปรแกรมเชิงวัตถุ

การเขียนโปรแกรมเชิงกระบวนการ (Procedural Programming)	การเขียนโปรแกรมเชิงวัตถุ (Object-Oriented Programming)
เน้นลำดับขั้นตอนกระบวนการ	เน้นวัตถุ
แยก data ออกจาก function	ผูก data กับ method เข้าด้วยกัน
ใช้ข้อมูลร่วมกัน	ซ่อนหรือแสดงรายละเอียดข้อมูล
การออกแบบโปรแกรมแบบ Top-down	การออกแบบโปรแกรมแบบ Bottom-up

1.2 Abstraction


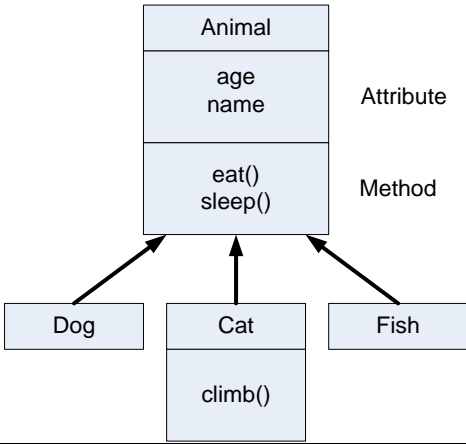
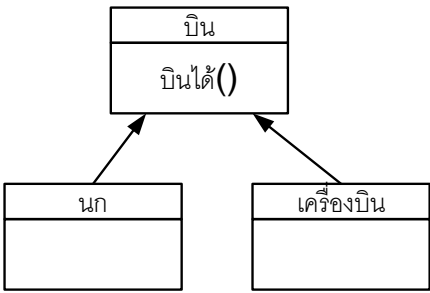
การคิดแบบ abstraction กล่าวคือ มนุษย์สามารถมองวัตถุที่มีความซับซ้อนด้วยการมองแบบ abstraction เช่น มนุษย์มองรถยนต์ คือ วัตถุรถยนต์ (พาหนะที่พาเราเคลื่อนที่จากจุดหนึ่งไปยังอีกจุดหนึ่ง) แต่จะไม่มองว่ารถยนต์ คือ วัตถุที่ประกอบด้วยชิ้นส่วนต่างๆ นับพันชิ้นและก็จะไม่พิจารณาลงรายละเอียดถึงระบบการเผาไหม้ ระบบเครื่องยนต์ ระบบเบรก หรือระบบทำงานต่างๆ ที่ซับซ้อนจนกลายเป็นรถยนต์ เมื่อมองจากภายนอกรถยนต์ คือ วัตถุหนึ่งเดียว แต่เมื่อมองเข้าไปข้างในจะพบว่ารถยนต์ประกอบด้วยระบบย่อยๆ หลายระบบหรือวัตถุย่อยๆ หลายวัตถุ เช่น ระบบเบรก ระบบไฟฟ้า ล้อ เพลา เครื่องเสียง เป็นต้น ในลักษณะ

แบบเดียวกันวัตถุเครื่องเสียงก็จะประกอบด้วย วิทย์ เครื่องเล่นซีดี เป็นต้น จากที่กล่าวมาจะพบว่ามนุษย์สามารถมองวัตถุที่มีความซับซ้อนในลักษณะลำดับชั้นก็ได้ และการมองในลักษณะลำดับชั้นนี้เรียกว่า Hierarchical abstractions

การเขียนโปรแกรมเชิงวัตถุได้พัฒนาบนแนวคิดแบบ abstraction และ hierarchical abstraction โดยการเขียนโปรแกรมเชิงวัตถุนั้นเกิดขึ้นมาเพื่อต้องการลดความซับซ้อนของโปรแกรม ทำให้การจัดการ แก้ไข การเรียกใช้ข้อมูลเป็นไปได้ง่าย รวมทั้งง่ายต่อการนำกลับมาใช้ใหม่ (reuse)

1.3 Object Class และคุณสมบัติของโปรแกรมเชิงวัตถุ

คุณสมบัติ	รายละเอียด
<p>Object</p> 	<p>วัตถุเป็นกุญแจสำคัญที่ทำให้เข้าใจเทคโนโลยีเชิงวัตถุ กล่าวคือ เมื่อมองไปรอบ ๆ จะพบตัวอย่างมากมายของวัตถุ เช่น สุนัข, โต๊ะ, โทรศัพท์, จักรยาน เป็นต้น</p> <p>วัตถุประกอบได้ด้วยคุณลักษณะ (attribute) และคุณสมบัติหรือการกระทำ (property or method)</p> <p>สุนัขมีคุณลักษณะ คือ ชื่อ, สี, พันธุ์, และการกระทำ คือ เหา่, กระดิกหาง</p> <p>คุณลักษณะและการกระทำของวัตถุเป็นวิธีคิดสำหรับการเขียนโปรแกรมเชิงวัตถุ</p>
<p>Class</p> 	<p>Class คือ นิยามหรือพิมพ์เขียวที่กำหนดไว้ล่วงหน้าก่อนจะเกิด objects หรือ instances โดยในพิมพ์เขียวจะประกอบไปด้วยกลุ่มของ attributes และ กลุ่มของ methods ที่แสดงค่าคุณลักษณะและการกระทำของ object</p>
<p>Abstraction</p> 	<p>การสร้างวัตถุแบบความคิดรวบยอด โดยการสร้างคลาสที่ประกอบไปด้วยคุณลักษณะ (attribute) และการกระทำ (method) ผู้ใช้ไม่จำเป็นต้องสนใจว่าคลาสนั้นมีรายละเอียดการทำงานอย่างไร ขอให้เรียกใช้งานได้อย่างถูกต้องผ่านทาง attribute และ method ก็พอ</p>
คุณสมบัติ	รายละเอียด

<p>Encapsulation</p> 	<p>การเขียนโปรแกรมลักษณะ OOP นั้น encapsulation ถือว่าเป็นองค์ประกอบหลักอย่างหนึ่ง ซึ่ง encapsulation หมายถึง การปกป้องและห่อหุ้ม attribute และ method เพื่อป้องกันข้อมูลหรือ method ของตนเองไม่ให้ถูกเข้าถึง โดยไม่ได้รับอนุญาตจาก object อื่นๆ</p> <p>ประโยชน์ของ encapsulation คือ ถ้าเราปล่อยให้ object อื่นเข้าถึงข้อมูลของเราได้อย่างอิสระข้อมูลที่เป็นความลับก็จะถูกเปิดเผยได้ หรือเมื่อมีการเปลี่ยนแปลงค่าข้อมูลก็จะทำให้ program ทำงานผิดพลาดได้ หรือเมื่อเราใช้ method อาจจะทำให้การเรียกใช้แบบผิดวิธี หรือผิดวัตถุประสงค์ได้</p>
<p>Inheritance</p> 	<p>คุณสมบัติการสืบทอด คือ คลาสลูกสามารถสืบทอด attributes และ methods จากคลาสแม่ได้ เปรียบเหมือน ลูกสาวสามารถสืบทอดรูปร่างหน้าตาและกิริยามารยาทมาจากแม่ตนเอง และคลาสลูกที่สืบทอดมานั้นยังสามารถพัฒนาต่อยอดได้ โดยเพิ่มเติมแก้ไข attributes หรือ methods ได้</p> <p>จากรูป class Animal เป็นคลาสแม่ ที่ประกอบไปด้วย attributes: age, name methods: eat(), sleep()</p> <p>Class Dog, Cat, Fish เป็นคลาสลูกที่สืบทอดมาจาก class Animal โดย class cat เพิ่ม method climb() เป็นต้น</p>
<p>Polymorphism</p> 	<p>การมีได้หลายรูปแบบ (many forms) เป็นคุณสมบัติที่ความสามารถของ method มีได้หลายรูปแบบขึ้นอยู่กับชนิดของวัตถุหรือพารามิเตอร์ที่ส่งไปทำงานของเมธอด เช่น คลาส บิน มีเมธอด บินได้() ซึ่งมีคลาส นก และคลาสเครื่องบิน เป็น subclass แต่ผลของการเรียกใช้เมธอด บินได้() ของอ็อบเจกต์จากคลาสทั้งสองจะออกมาไม่เหมือนกัน</p>

1.4 เริ่มต้นเขียนโปรแกรมภาษา java ด้วยภาษา C

หัวข้อ	ภาษา C	ภาษา Java
คำสั่งข้อมูลนำเข้า Scan	<pre>int i; scanf("%d", &i);</pre>	<pre>Scanner scan = new Scanner(System.in); int i = scan.nextInt();</pre> รายละเอียด methods แสดงดังตาราง 1.2
คำสั่งแสดงผล print	<pre>printf("%d\n", i);</pre>	<pre>System.out.println(i);</pre>
รูปแบบการเขียน โปรแกรม main	<pre>#include<stdio.h> void main() { int i; scanf("%d", &i); printf("%d", i); }</pre>	<pre>import java.util.Scanner; class TestI { public static void main(String args[]) { Scanner scan = new Scanner(System.in); int i = scan.nextInt(); System.out.print(i); } }</pre>
Program file	xxx.c	TestI.java และ TestI.class

ตาราง 1.2 แสดง methods for Scanner Object

Method	คำอธิบาย
nextByte()	อ่านข้อมูลจำนวนเต็มชนิด byte
nextShort()	อ่านข้อมูลจำนวนเต็มชนิด short
nextInt()	อ่านข้อมูลจำนวนเต็มชนิด int
nextLong()	อ่านข้อมูลจำนวนเต็มชนิด long
nextFloat()	อ่านข้อมูลจำนวนจริงชนิด float
nextDouble()	อ่านข้อมูลจำนวนจริงชนิด double

1.5 แบบฝึกหัดท้ายบท

- จงอธิบายถึงความแตกต่างระหว่าง attribute และ method
- จงอธิบายถึงความแตกต่างระหว่าง class และ object
- จงสร้าง class mobile ที่ประกอบไปด้วย 10 attributes และ 10 methods
- จงอธิบายความสัมพันธ์ระหว่าง class แม่ และ class ลูก พร้อมทั้งยกตัวอย่างแสดง class ลูก ของ class mobile จากข้อ 3
- จงนิยาม Class ชื่อ Circle2D ที่ประกอบไปด้วย
 - Attribute 2 จำนวน คือ x, y ซึ่งแสดงค่าจุดศูนย์กลางของวงกลม
 - Attribute radius ซึ่งแสดงค่ารัศมีของวงกลม
 - Method getArea() ที่ return ค่าพื้นที่ของวงกลม
 - Method getPerimeter() ที่ return เส้นรอบวง
- จงหาผลลัพธ์ของโปรแกรม

class Binomial		ผลลัพธ์
1	class Binomial	
2	{	
3	public static void main(String[] args){	
4	int N = 7;	
5	int K = 5;	
6	long[][] binomial = new long[N+1][K+1]; //Array 2D	
7		
8	for (int k = 1; k <= K; k++)	
9	binomial[0][k] = 0;	
10	for (int n = 0; n <= N; n++)	
11	binomial[n][0] = 1;	
12	for (int n = 1; n <= N; n++)	
13	for (int k = 1; k <= K; k++)	
14	binomial[n][k] = binomial[n-1][k-1] + binomial[n-1][k];	
15		
16	System.out.println(binomial[N][K]);	
17	}	
18	}	

7. จงหาผลลัพธ์ของโปรแกรม

class XX		ผลลัพธ์
1	class XX	
2	{	
3	public static void main(String[] args){	
4	int g = f(4, 2, 5, 9);	
5	System.out.println(g);	
6		
7	static int f(int m, int n)	
8	{	
9	return 3;	
10	}	
11	static int f(int m, int n, int o)	
12	{	
13	return 12;	
14	}	
15	}	
16	}	

8. จงหาผลลัพธ์ของโปรแกรม

SomethingIsWrong		ผลลัพธ์
1	class SomethingIsWrong	
2	{	
3	public static void main(String[] args){	
4	Rectangle myRect;	
5	myRect.width = 40;	
6	myRect.height = 50;	
7	System.out.println("myRect area is "+ myRect.area());	
8	}	
9	}	

โจทย์เพิ่มทักษะการเขียนโปรแกรม

จงเขียนโปรแกรมเพื่อหาผลลัพธ์ของปัญหาข้อ 9 – 12 ทั้งแบบภาษา C และภาษา java

9. จงเขียนโปรแกรมที่อ่านค่า Fahrenheit degree แบบ double จาก Keyboard จากนั้นให้แปลงเป็น Celsius โดย $32 \leq \text{Fahrenheit} \leq 4000$ และแสดงผลลัพธ์ทศนิยม 4 ตำแหน่ง สูตรของการคำนวณคือ $\text{Celsius} = (5/9) * (\text{Fahrenheit} - 32)$

10. Assume a runner runs 14 kilometers in 45 minutes and 30 seconds. Write a program that displays the average speed in miles per hour. (Note that 1 mile is 1.6 kilometers)

11. Assume a runner runs 24 miles in 1 hour, 40 minutes, and 35 seconds. Write a program that displays the average speed in kilometers per hour. (Note that 1 mile is 1.6 kilometers)

12. The U.S. Census Bureau projects population based on the following assumptions:

- One birth every 7 seconds
- One death every 13 seconds
- One new immigrant every 45 seconds

Write the program to display the population for each of the next five year. Assume the current population is 312,032,486 and one year has 365 days. (Hin: In Java, if two integers perform division, the result is an integer. The fraction part is truncated. For example, $5/4$ is 1 (not 1.25) and $10/4$ is 2 (not 2.5))

1. จงเขียนโปรแกรมคำนวณสมการ $3x^4 + 2x^3 - x + 10$ โดย x เป็นจำนวนเต็มที่ได้รับค่าผ่าน keyboard

2. เขียนโปรแกรมคำนวณค่าแรง ถ้าทำงาน 40 ชม./สัปดาห์ ได้ค่าแรง ชั่วโมงละ 120 บาท แต่ถ้าทำงานเกิน 40 ชม./สัปดาห์ จะได้ค่า O. T. 20 บาท/ชม.

3. กำหนดให้ [23, 16, 27] จงเขียนโปรแกรมหาค่ามากที่สุดห้ามใช้คำสั่ง for loop

4. กำหนดให้ [23, 58, 88, 16, 27] จงเขียนโปรแกรมหาค่ามากที่สุดห้ามใช้คำสั่ง for loop

5. Write a program that read an integer between 100 and 999, and find the reverse of that integer. For Example, if an integer is 123, its reverse is 321