

JAVA

3
IN
1

เขียนจาวาครบเครื่อง

Java Basic: จาวาพื้นฐาน ครอบคลุม จาวา 15

Java OOP: การเขียนโปรแกรมเชิงวัตถุด้วยภาษาจาวา

Java GUI: จาวากราฟิกอินเทอร์เฟซ (swing) จาก NetBeans

โดย ศูนย์หนังสือราคานักเรียน
450 บาท

คำนำ

วัตถุประสงค์หลักหนังสือเล่มนี้ นำเสนอเนื้อหาให้ครอบคลุมสำหรับผู้ศึกษาการเขียนโปรแกรมภาษาจาวาใน 3 เนื้อหาหลัก ได้แก่

1. จาวาพื้นฐาน ได้แก่ โครงสร้างโปรแกรมภาษาจาวา โครงสร้างและไวยากรณ์ การประกาศและการใช้ตัวแปร นิพจน์และการควบคุมลำดับการทำงาน การนำเข้าและการแสดงผล อาร์เรย์ และสตริง
2. จาวาแบบเชิงวัตถุ ได้แก่ แนวคิดเชิงวัตถุ คลาส การสร้างและการใช้งาน อ็อบเจกต์ การสร้างและการใช้งานฟังก์ชัน การห่อหุ้ม การสืบทอด การพ้องรูป การจัดการกับภาวะผิดปกติ และเชรด
3. การเขียนโปรแกรมภาษาจาวาด้วยจาวาสวิงซ์ โดยอาศัยโปรแกรมเนตบีน (NetBeans) ได้แก่ เนื้อหาการใช้งาน เจเฟรม (JFrame) และ เจคอมโพเนนต์ (JComponent) อื่นๆ สำหรับสร้างแอปพลิเคชันใช้งานด้วยภาษาจาวา

ทั้งนี้ ด้วยเนื้อหาที่นำเสนอทั้งหมดเหมาะสำหรับนักเรียน นักศึกษา และครูอาจารย์ ผู้ต้องการศึกษาการเขียนโปรแกรมและนำไปใช้เรียนใช้สอนในรายวิชาปฏิบัติการเขียนโปรแกรม ในเนื้อหาของเล่มนี้ จึงเน้นเนื้อหาปฏิบัติที่นำเสนอด้วยรหัสโปรแกรมภาษาจาวา ผลของการทำงานของโปรแกรม และรวมไปถึงการบูรณาการฝึกปฏิบัติสู่การประยุกต์เป็นแอปพลิเคชันอีกด้วย

ผู้เขียนหวังว่าผู้ศึกษาจากหนังสือเล่มนี้ จะได้ประโยชน์ตามวัตถุประสงค์ที่ตั้งเอาไว้ และมีความรู้ความเข้าใจสำหรับนำเนื้อหาความรู้ไปสู่การพัฒนาตนเองเป็นนักวิทยาการคอมพิวเตอร์หรือนักเทคโนโลยีคอมพิวเตอร์ที่สมบูรณ์ต่อไป

ศูนย์หนังสือราคานักเรียน

14 กรกฎาคม 2564

สารบัญ

หน้า

บทที่ 1	แนะนำโปรแกรมภาษาจาวา	1
	ความรู้เบื้องต้นเกี่ยวกับภาษาจาวา	1
	สถิติความนิยมของใช้งานโปรแกรมภาษาจาวา	1
	ประวัติของโปรแกรมภาษาจาวา	3
	วิวัฒนาการทางด้านความสามารถของภาษาจาวา	5
	การติดตั้งโปรแกรมภาษาจาวาและเครื่องมือพัฒนาโปรแกรม	9
	การพัฒนารหัสต้นฉบับของภาษาจาวา	9
	การดาวน์โหลด	10
	การติดตั้งภาษาจาวา	11
	โครงสร้างและรูปแบบพื้นฐานของภาษาจาวา	12
	การพัฒนาโปรแกรมภาษาจาวาด้วยเน็ตเบินไอดีอี	18
	สรุป	20
	แบบฝึกหัดท้ายบท	21
บทที่ 2	คำสั่งพื้นฐานของภาษาจาวา	22
	การแสดงผลมาตรฐาน	22
	คำสั่ง System.out.print	22
	คำสั่ง System.out.println	25
	ลำดับหลุดพ้น (Escape Sequence)	26
	การจัดรูปแบบการแสดงผล	28
	การใช้งานคำสั่ง System.out.format	28
	การใช้งานคำสั่ง System.out.printf	31
	การจัดรูปแบบเลขทศนิยม	32
	คำสั่งนำเข้าข้อมูล	34
	หลักการนำเข้าข้อมูลเบื้องต้น	34
	ตัวแปรและการประกาศตัวแปร	34
	ชนิดของตัวแปร	35
	การกำหนดสิทธิการเข้าถึงข้อมูล	36
	การนำเข้าข้อมูลด้วย BufferedReader	37
	การนำเข้าข้อมูลด้วย Scanner	44
	การรับข้อมูลด้วย JOptionPane	47
	ตัวดำเนินการของภาษาจาวา	49

ความหมายและประเภทของตัวดำเนินการ	49
ตัวดำเนินการกำหนดค่า	49
ลำดับขั้นตอนการดำเนินการ	50
ตัวดำเนินการทางคณิตศาสตร์	51
ตัวดำเนินการเปรียบเทียบความสัมพันธ์และตรรกะ	67
ตัวดำเนินการบิตไวส์	71
สรุป	74
แบบฝึกหัดท้ายบท	76
บทที่ 3 คำสั่งควบคุมลำดับการทำงานของภาษาจาวา	79
คำสั่งเงื่อนไข	80
ความหมายและความสำคัญของคำสั่งเงื่อนไข	80
คำสั่ง if	80
คำสั่ง switch	92
คำสั่งวนซ้ำ	97
ความหมายและประเภทของคำสั่งวนซ้ำ	97
คำสั่ง for	97
คำสั่ง while	103
คำสั่ง do ... while	108
สรุป	113
แบบฝึกหัดท้ายบท	114
บทที่ 4 อาร์เรย์ สตริง และแฟ้มข้อมูล	117
อาร์เรย์	118
ความหมายและความสำคัญของอาร์เรย์	118
อาร์เรย์ 1 มิติ	118
อาร์เรย์ 2 มิติ	122
อาร์เรย์ 3 มิติ	129
สตริง	133
ความรู้พื้นฐานเกี่ยวกับสตริง	133
การดำเนินการกับสตริง	136
แฟ้มข้อมูล	152
ความสำคัญและประเภทของแฟ้มข้อมูล	152
เท็กซ์ไฟล์	152
แฟ้มแบบสุ่ม	162
สรุป	174
แบบฝึกหัดท้ายบท	176

บทที่ 5 ความรู้เบื้องต้นเกี่ยวกับการเขียนโปรแกรมเชิงวัตถุ.....	179
แนวคิดการเขียนโปรแกรมด้วยภาษาเชิงวัตถุ.....	180
ส่วนประกอบเชิงวัตถุ.....	181
คลาส.....	183
การสร้างคลาสและการสร้างวัตถุ.....	185
การสืบทอดคุณสมบัติ.....	188
การโอเวอร์โหลด.....	194
การโอเวอร์ไรต์.....	196
สัญลักษณ์และแผนภาพของยูเอ็มแอล.....	197
คลาสไดอะแกรม.....	199
ความสัมพันธ์ระหว่างคลาส.....	206
การสื่อสารระหว่างวัตถุ.....	206
รูปแบบของความสัมพันธ์ระหว่างคลาส.....	207
การเขียนการสืบทอดคุณสมบัติ.....	212
อินเฮอริเทนซ์.....	212
อินเทอร์เฟซ.....	215
สรุป.....	218
แบบฝึกหัดท้ายบท.....	219
บทที่ 6 คลาส เมธอด อินเทอร์เฟซ และแอ็บสแตรกคลาส.....	221
การสร้างคลาสและการสร้างวัตถุ.....	222
การใช้งานเมธอด.....	231
ความหมายและความสำคัญของเมธอด.....	231
เมธอดที่ใช้งานปกติ.....	232
เมธอดแบบไม่คืนค่า.....	232
เมธอดไม่คืนค่าแบบเรียกใช้โดยผ่านค่าพารามิเตอร์.....	235
เมธอดแบบคืนค่า.....	237
คอนสตรัคเตอร์.....	241
การเข้าถึงแอตทริบิวต์.....	242
โอเวอร์โหลดฟังก์ชัน.....	244
อินเทอร์เฟซ.....	247
แนวคิดของการสร้างอินเทอร์เฟซ.....	247
การใช้งานอินเทอร์เฟซแบบเดี่ยว.....	253
การใช้งานอินเทอร์เฟซร่วมกัน.....	264
การใช้งานอินเทอร์เฟซร่วมกับการสืบทอด.....	274
แอ็บสแตรกคลาส.....	279

สรุป	282
แบบฝึกหัดท้ายบท	283
บทที่ 7 ความสัมพันธ์ของคลาสและการสืบทอด	285
ความสัมพันธ์แบบ Dependency	286
ความสัมพันธ์แบบ Aggregation	295
ความสัมพันธ์แบบ Composition	307
ความสัมพันธ์แบบ Association	310
แนวทางการใช้งานการสืบทอดคุณสมบัติ	318
หลักการอินเฮอริเทน	319
การเรียกใช้งานคอนสตรัคเตอร์ super และอ้างอิง this	336
การเข้าถึงแอตทริบิวต์และเมธอด	341
โอเวอร์ไรด์	347
สรุป	349
แบบฝึกหัดท้ายบท	350
บทที่ 8 โพลีมอร์ฟิซึมและดีไซด์แพทเทิร์น	351
โพลีมอร์ฟิซึม	351
เครื่องมือสำหรับสร้างการใช้งานโพลีมอร์ฟิซึม	356
การกำหนดตัวแปรแบบยึดติดและแบบพลวัต	362
การสร้างโพลีมอร์ฟิซึมโดยโอเวอร์โหลดดิ่ง	371
การสร้างโพลีมอร์ฟิซึมด้วยการสืบทอด	377
ดีไซด์แพทเทิร์น	387
ครีเอชันดีไซด์แพทเทิร์น	389
สตรักเจอร์ดีไซด์แพทเทิร์น	397
บีเฮฟวิเออร์ดีไซด์แพทเทิร์น	405
เฟรมเวิร์ก	407
สรุป	410
แบบฝึกหัดท้ายบท	412
บทที่ 9 การดักจับข้อผิดพลาดและเทรต	413
การดักจับข้อผิดพลาด	413
ความหมายและความสำคัญของการดักจับข้อผิดพลาด	413
รูปแบบของการดักจับ	415
การสร้างการดักจับข้อผิดพลาด	425
การจัดข้อผิดพลาด	427
ข้อผิดพลาดในแพ็คเกจที่ใช้งานเป็นประจำ	432
เทรต	433

แนวคิดเกี่ยวกับเทรด	435
การใช้งานเทรดด้วยภาษาจาวา	436
สรุป	441
แบบฝึกหัดท้ายบท	443
บทที่ 10 จาวาสวิงซ์.....	444
การใช้งานจียูไอด้วยเจเฟรมและเจพาเนล	447
เจเฟรมฟอร์ม	447
เจพาเนล	455
เครื่องมือรับและแสดงผล	458
เจเลเบิล	458
เจเท็กซ์ฟิลด์	461
เจฟอร์มเมตเทดเท็กซ์ฟิลด์	465
เจบัททอน	467
เจทอกเกิลบัททอน	470
เจเท็กซ์อาเรีย	471
เจพาสเวิร์ดฟิลด์	472
เครื่องมือสนับสนุนการรับและแสดงผล	474
เจคอมโบบ็อกซ์	474
เจลิส	477
เจเช็คบ็อกซ์	479
เจเรดิโอบัททอนและเจบัททอนกรุป	480
เจโพรเกรสบาร์	483
เจสครอลบาร์	485
เจสไลเดอร์	486
เจสปินเนอร์	489
เจแท็บเบดเพน	490
เจสปลิทเพน	493
เจเดสก์ท็อพเพนและเจอินเทอร์นอลเฟรม	494
เครื่องมือโต้ตอบและจัดรูปแบบข้อมูล	497
เจไดอะล็อก	497
เจไฟล์ชู้สเซอร์	500
เจทรี	505
เจคัลเลอร์ชู้สเซอร์	508
เจอ็อบชันเพน	509
เจเทเบิล	512

เครื่องมือสร้างเมนู.....	517
เมนูบาร์.....	517
เมนูลบริ.....	521
เมนูแอป.....	523
สรุป.....	524
แบบฝึกหัดท้ายบท.....	525
บรรณานุกรม.....	529

บทที่ 1

แนะนำโปรแกรมภาษาจาวา

(Introduction to Java Programming Language)

วัตถุประสงค์

เพื่อให้ผู้เรียน

1. มีความรู้และอธิบายเกี่ยวกับความหมายและความจำเป็นของโปรแกรมภาษาจาวา
2. มีความรู้ ประวัติและวิวัฒนาการ แนวคิด การพัฒนาโปรแกรมด้วยภาษาจาวา
3. มีความรู้เบื้องต้นเกี่ยวกับภาพรวมของโปรแกรมภาษาจาวา
4. มีความรู้การพัฒนาโปรแกรมภาษาจาวาเบื้องต้น

โปรแกรมภาษาจาวาเป็นโปรแกรมภาษาคอมพิวเตอร์ที่สนับสนุนแนวคิดเชิงวัตถุ ได้รับความนิยมสูงสุดในระยะเวลากว่า 20 ปีที่ผ่านมา และปัจจุบันก็ยังถูกจัดให้เป็นโปรแกรมภาษาในอันดับ 1 ของการพัฒนาโปรแกรมโดยทั่วไปและบนระบบคลาวด์ แอปพลิเคชันของภาษาจาวาจะถูกคอมไพล์เป็นไบนารีโค้ด (bytecode) ซึ่งสามารถทำงานบนเครื่องเสมือน(JVM) โดยไม่คำนึงถึงสถาปัตยกรรมคอมพิวเตอร์_ไวยากรณ์ของจาวามีความคล้ายคลึงกับภาษาซีและซีพลัสพลัส ดังเดิมพัฒนามาโดยเจมส์กอสลิงที่ซันไมโครซิสเต็มส์ เปิดตัวในปี 1995 ต่อมาถูกซื้อกิจการโดยบริษัทโอราเคิลในปี 2007 และพัฒนาอย่างต่อเนื่อง จนกระทั่งในเดือนกันยายน 2020 เวอร์ชันล่าสุดคือ Java 15_เนื่องจากเป็นโปรแกรมภาษาที่สามารถพัฒนาโปรแกรมได้หลายแพลตฟอร์ม (Platform) โดยเขียนโปรแกรมเพียงครั้งเดียว แล้วสามารถนำไปใช้งานได้บนระบบปฏิบัติการที่แตกต่างกันได้ โปรแกรมภาษานี้ให้ดาวน์โหลดได้โดยไม่มีค่าใช้จ่าย อีกทั้งเป็นต้นแบบของการเขียนโปรแกรมแบบเชิงวัตถุได้เป็นอย่างดี

บทนี้นำเสนอเนื้อหาเกี่ยวกับโปรแกรมภาษาจาวาโดยเริ่มจากการพัฒนาโปรแกรมภาษาจาวาและแนวคิดเบื้องต้น การดาวน์โหลดตัวแปลภาษา การติดตั้ง การใช้งานเบื้องต้น แนวทางการพัฒนารหัสต้นฉบับด้วยคลาส การคอมไพล์และการรันโปรแกรม โดยเน้นให้ผู้อ่านได้เข้าใจหลักการเบื้องต้นของการเขียนโปรแกรมด้วย

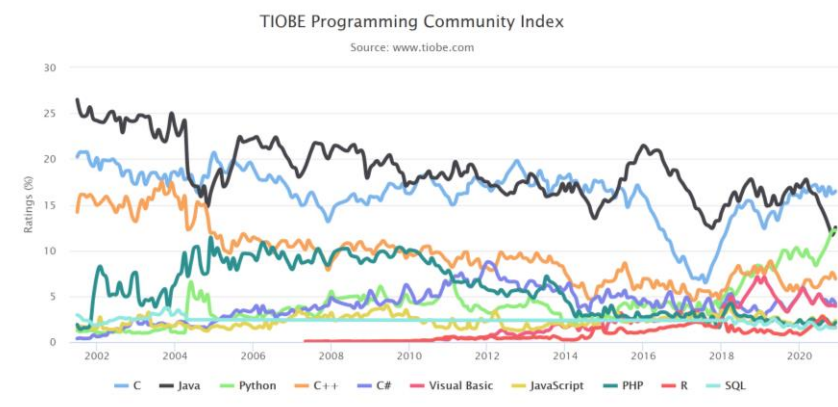
ความรู้เบื้องต้นเกี่ยวกับภาษาจาวา

สถิติความนิยมของใช้งานโปรแกรมภาษาจาวา

สำหรับการตัดสินใจในการศึกษาโปรแกรมภาษาใดภาษาหนึ่ง จำเป็นต้องรู้จักว่าสิ่งที่กำลังจะเรียนรู้นั้นคุ้มค่าต่อการศึกษาหรือไม่ สำหรับผู้ที่ตัดสินใจศึกษาโปรแกรมภาษาจาวา ในตำราเล่มนี้ของนำเสนอสถิติต่างๆเกี่ยวกับโปรแกรมภาษาจาวา ที่ได้ศึกษาค้นคว้ามาจากอินเทอร์เน็ต ซึ่งจะแสดงให้เห็นถึงความนิยมของการใช้งานโปรแกรมภาษาจาวา จาก <https://www.tiobe.com/tiobe-index/>

Dec 2020	Dec 2019	Change	Programming Language	Ratings	Change
1	2	▲	C	16.48%	+0.40%
2	1	▼	Java	12.53%	-4.72%
3	3		Python	12.21%	+1.90%
4	4		C++	6.91%	+0.71%
5	5		C#	4.20%	-0.60%
6	6		Visual Basic	3.92%	-0.83%
7	7		JavaScript	2.35%	+0.26%
8	8		PHP	2.12%	+0.07%
9	16	▲	R	1.60%	+0.60%
10	9	▼	SQL	1.53%	-0.31%

ภาพที่ 1.1 แสดงการจัดลำดับโปรแกรมภาษาจาก <https://www.tiobe.com/tiobe-index/>



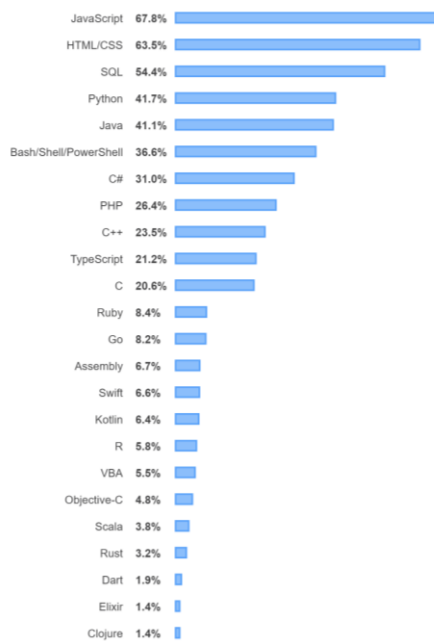
ภาพที่ 1.2 แสดงการจัดลำดับโปรแกรมภาษาจาก <https://www.tiobe.com/tiobe-index/> เป็นกราฟ

จากการจัดลำดับของ <https://www.northeastern.edu/graduate/blog/most-popular-programming-languages/> จัดลำดับตามรายได้ของผู้ที่ใช้งานและมีความรู้ ปี 2020 จาวยอยู่ 3 ในลำดับที่มีงาน 29,000 เงินที่ผู้ทำงานจะได้รับต่อปี ประมาณ 104,000 ดอลลาร์สหรัฐอเมริกา แสดงดังรูปที่ 1.3

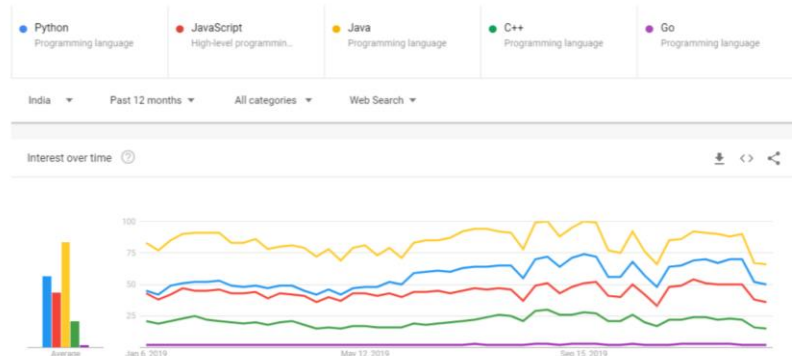


ภาพที่ 1.3 แสดงการจัดลำดับโปรแกรมภาษาจาก <https://www.northeastern.edu/graduate/blog/most-popular-programming-languages/>

นอกจากนั้นในการสำรวจโดย <https://insights.stackoverflow.com/survey/2019#technology> ในปี 2019 สำหรับการพัฒนาโปรแกรมโดยทั่วไป แสดงดังรูปที่ 1.4 (ก) และการสำรวจของ <https://www.geeksforgeeks.org> แสดงดังรูปที่ 1.4 (ข) จาก <https://www.geeksforgeeks.org/top-10-programming-languages-to-learn-in-2020-demand-jobs-career-growth/>



(ก)



(ข)

ภาพที่ 1.4 แสดงการจัดลำดับโปรแกรมภาษาจาก <https://insights.stackoverflow.com> และ <https://www.geeksforgeeks.org>

สำหรับในประเทศไทย <https://www.asearcher.com/> ก็ยังจัดให้จาวาเป็นภาษายอดนิยมอันดับหนึ่งในประเทศไทย <https://www.joblnw.com/article/programmer-highest-demand> จัดให้เป็นลำดับดังนี้ ลำดับงาน React, Java, Swift, Kotlin, PHP ในขณะที่ <https://www.blognone.com/node/98631> อันดับภาษาที่การจ้างงานสูงในปี 2018 จาวาก็ยังเป็นอันดับหนึ่ง <https://dailycode.cc/programming-language-ที่น่าลุยในปี-2020> จัดให้ Python, JavaScript, Java ซึ่งจะพบว่าในรอบปีที่ผ่านมาโปรแกรมภาษาจาวาได้รับความนิยมสูงสุดมาตลอดถึงหาเทียบกับโปรแกรมภาษาพื้นฐานเดิมเช่น C, C#, C++, Visual Basic โปรแกรมภาษาจาวาจะได้รับความนิยมเป็นอันดับหนึ่งเสมอ ดังนั้นจากเหตุผลการจัดลำดับดังกล่าวที่ผ่านมา จึงเป็นเหตุผลว่าการศึกษโปรแกรมภาษาจาวามีความจำเป็นอย่างไร

ประวัติของโปรแกรมภาษาจาวา

ภาษาจาวา พัฒนามาจากภาษาโอ๊ก (Oak) ในราวปี ค.ศ. 1990 โดยทีมพัฒนาจากบริษัทซันไมโครซิสเต็ม (Sun Microsystems) เพื่อใช้ฝังตัวกับอุปกรณ์อิเล็กทรอนิกส์ ต่อมามีการใช้อินเตอร์เน็ตอย่างแพร่หลาย ทีมพัฒนาจึงมีแนวคิดในการพัฒนาโปรแกรมให้สามารถใช้งานได้กับทุก ๆ แพลตฟอร์มโดยไม่ต้องเขียนโปรแกรมใหม่ และแนวคิดนี้ได้พัฒนาจนสำเร็จใช้งานได้จริง ในราวปี ค.ศ. 1995 จึงเปลี่ยนชื่อจาก Oak เป็น Java พร้อมกันนี้บริษัทอนุญาตให้ดาวนโหลดเครื่องมือที่ใช้ในการพัฒนา เรียกว่า จาวาดีเวลลอปเม้นต์คิต (Java Development Kit) หรือเจดีเค (JDK) โดย Java 1.0 นำมาออกมาในปี 1996 ซึ่งในปัจจุบันพัฒนาถึงเวอร์ชัน 15 (มกราคม 2021) การทำงานของภาษาจาวาจะพัฒนารหัสต้นฉบับ (Source code) เหมือนกับโปรแกรมภาษาอื่น ๆ หลังจากนั้นคอมไพล์ (Compile) เป็นแฟ้มไบต์โค้ด (Byte Code) สำหรับใช้กับทุกแพลตฟอร์ม เมื่อต้องการจะรันสำหรับแต่ละแพลตฟอร์มจะใช้จาวาเวอร์ชวลแมชีน (Java Virtual Machine) ทำหน้าที่แปลไบต์โค้ดเพื่อใช้งาน ซึ่งคุณสมบัติดังกล่าว ทำให้ภาษาจาวาได้ชื่อว่าเป็นภาษาที่เขียนเพียงครั้ง

เดียวแล้วสามารถไปใช้งานได้ทุก ๆ แพลตฟอร์ม (Write Once Run Anywhere) ภาษาจาวามีอยู่หลากหลายแพลตฟอร์ม เช่น จาวาแอปพลิเคชัน (Java Application) สำหรับเขียนโปรแกรมแบบแอปพลิเคชัน จาวาแอ็บเพล็ต (Java Applet) สำหรับพัฒนาโปรแกรมแอปพลิเคชันที่ใช้งานบนเว็บ, จาวาเซิร์ฟเล็ต (Java Servlet) สำหรับพัฒนาโปรแกรมบนเว็บ จาวาเซิร์ฟเวอร์เพจ (Java Server Page) เป็นภาษาสคริปต์ของจาวาสำหรับพัฒนางานบนเว็บ และเป็นภาษาจาวาสำหรับพัฒนาโปรแกรมบนระบบมือถืออีกด้วย เป็นต้น แต่ปัจจุบันนี้บริษัทโอราเคิล (Oracle) ได้ซื้อกิจการของบริษัทซันไมโครซิสเต็ม ในปี 2009 ซึ่งก็ทำให้ภาษาจาวาตกไปอยู่ในการดูแลของโอราเคิลด้วย โปรแกรมภาษาจาวาในปัจจุบัน ลำดับของเวอร์ชันต่างๆ มีดังนี้

ตารางที่ 1.1 แสดงเวอร์ชันของภาษาจาวา

เวอร์ชัน	วันที่
JDK เบต้า	พ.ศ. 2538
JDK1.0	23 มกราคม 2539
JDK 1.1	19 กุมภาพันธ์ 2540
J2SE 1.2	8 ธันวาคม 2541
J2SE 1.3	8 พฤษภาคม 2543
J2SE 1.4	6 กุมภาพันธ์ 2545
J2SE 5.0	30 กันยายน 2547
Java SE 6	11 ธันวาคม 2549
Java SE 7	28 กรกฎาคม 2554
Java SE 8	18 มีนาคม 2557
Java SE 9	21 กันยายน 2560
Java SE 10	20 มีนาคม 2561
Java SE 11	25 กันยายน 2561
Java SE 12	19 มีนาคม 2019
Java SE 13	17 กันยายน 2019
Java SE 14	17 มีนาคม 2020
Java SE 15	15 กันยายน 2020

ปัจจุบันภาษาจาวาพัฒนาถึงเวอร์ชัน 15 (ธันวาคม 2563) ซึ่งอยู่ในช่วงการพัฒนา ตำราเล่มนี้จะเน้นใช้งานที่เวอร์ชัน 8-15 ซึ่งจะใช้งานร่วมกับโปรแกรมเนทีฟ 8.2-12.0 รายละเอียดของโปรแกรมภาษาจาวาสามารถเข้าไปศึกษารายละเอียดต่าง ๆ ได้ที่ <http://www.oracle.com/technetwork/java/index.html>

วิวัฒนาการทางความสามารถของภาษาจาวา

ข้อมูลโดยตรงจาก โอราเคิล (29 ธันวาคม 2563) https://www.oracle.com/java/moved-by-java/timeline/?source=ow:o:p:po::RC_WWMK200728P00026:DevLiveJava_PromoOcomJava&intcmp=ow:o:p:po::RC_WWMK200728P00026:DevLiveJava_PromoOcomJava
ระบุเหตุการณ์สำคัญของภาษาจาวาดังนี้

พ.ศ. 2534–2537

- Sun Microsystems เริ่มทำงานกับภาษาโปรแกรม Oak ซึ่งเป็นส่วนหนึ่งของโครงการสีเขียว
- ทีม Green Project สาธิตคอมพิวเตอร์แท็บเล็ต Star7
- Duke มาสคอต Java ได้รับการออกแบบโดย Joe Palrang ผู้อำนวยการฝ่ายศิลป์สำหรับแอนิเมชันสาธิต Star7 Palrang ทำงานให้กับ DreamWorks ในภาพยนตร์การ์ตูนเช่น Shrek และ Antz
- เพื่อแสดงให้เห็นถึงเทคโนโลยีของ Oak และสร้างยานพาหนะที่จะนำซอฟต์แวร์ไปไว้ในมือให้ได้มากที่สุด ทีมงาน Green Project ได้พัฒนาเว็บเบราว์เซอร์ WebRunner ของตัวเองซึ่งได้รับการขนานนามว่าเป็นการแสดงความเคารพต่อภาพยนตร์ Blade Runner เบราว์เซอร์ถูกสร้างขึ้นโดยใช้ภาษาโปรแกรม Oak และเรียกใช้แอปพลิเคชัน Oak

พ.ศ. 2538

- ในการเตรียมเสนอขายครั้งแรกชั้นพยายามจดทะเบียน Oak เป็นชื่อผลิตภัณฑ์ แต่ชื่อถูกใช้ไปแล้ว ทางเลือกอื่นที่พิจารณาระหว่างการระดมความคิด ได้แก่ Silk, Lyric, Pepper, NetProse, Neon, Ruby, WebDancer และ WebSpinner การตรวจสอบทางกฎหมายได้กำจัดชื่อเหล่านี้ส่วนใหญ่ออกไป แต่มีอยู่ชื่อหนึ่ง: Java หลังจากนั้นไม่นานภาษาการเขียนโปรแกรม Java ก็มีโลโก้ถ้วยและไอน้ำที่น่าอัศจรรย์ในขณะนี้
- San Jose Mercury News จัดทำบทความหน้าหนึ่งซึ่งครอบคลุมเทคโนโลยี Java
- เบราว์เซอร์ WebRunner ซึ่งภายหลังเปลี่ยนชื่อเป็น HotJava แสดงให้เห็นเป็นครั้งแรกในการประชุม TED
- Java 1.0a2 ได้รับการประกาศในการประชุม SunWorld ซึ่งผู้นำของ Netscape ได้ประกาศการสนับสนุนเบราว์เซอร์สำหรับ Java Netscape อนุญาตให้ใช้สิทธิ์ Java และเผยแพร่เบราว์เซอร์ Netscape Navigator เวอร์ชันแรกที่เปิดใช้งาน Java
- Sun ขอแนะนำสโลแกน “Write Once, Run Anywhere” เพื่ออธิบายความสามารถข้ามแพลตฟอร์มที่เป็นเอกลักษณ์ของ Java

พ.ศ. 2539

- Java Development Kit 1.0 เปิดตัวแล้ว

- แนะนำระบบปฏิบัติการจาวา (JavaOS) JavaOS เขียนด้วย Java เป็นหลักและมีไว้สำหรับคอมพิวเตอร์เครือข่ายและระบบฝังตัว
- Sun เปิดตัวโปรแกรมการมีส่วนร่วมที่เน้นนักพัฒนาเป็นครั้งแรก: Java Developer Connection
- Sun จัดการประชุม JavaOne ครั้งแรกในซานฟรานซิสโก
- หนังสือการ์ตูน The Amazing Adventures of Duke วางจำหน่ายแล้ว หนังสือการ์ตูนความยาว 16 หน้า พากย์เสียงมาสดอคอต Java ว่า “Net's Smoothest Code Man”

พ.ศ. 2540–2543

- Java Development Kit 1.1 เปิดตัวพร้อมคุณสมบัติหลักของคลาสภายใน, JavaBeans, RMI Compiler, Reflection, คอมไพเลอร์ just-in-time (JIT), การทำให้เป็นสากลและการสนับสนุน Unicode
- Java Development Kit 1.2 ได้รับการเผยแพร่และเปลี่ยนชื่อเป็นแพลตฟอร์ม Java 2, Standard Edition (J2SE 1.2) คุณสมบัติใหม่ ได้แก่ Swing, Java 2D และ Collections Framework
- Java Community Process (JCP) ก่อตั้งขึ้นเพื่อทำให้เป็นทางการกลไกที่อนุญาตให้ผู้สนใจช่วยพัฒนาข้อกำหนดทางเทคนิคสำหรับเทคโนโลยี Java
- J2SE 1.3 เปิดตัวพร้อมคุณสมบัติใหม่ทั้งหมด HotSpot Java VM, Java Naming และ Directory Interface และ Java Platform Debugger Architecture
- สมาร์ทการ์ดที่รองรับ Java มากกว่า 100 ล้านการ์ดซึ่งใช้ Java Card (เปิดตัวในปี 1996) จะถูกจัดส่งภายในสิ้นปี 2000
- เปิดตัวต้นแบบเครื่องเล่นออปติคัลดิสก์ Blu-ray ที่รองรับ Java เครื่องแรก

พ.ศ. 2544–2547

- Larry Ellison CEO ของ Oracle เข้าร่วมเป็นผู้นำ Sun บนเวทีที่ JavaOne เพื่อสาธิตซอฟต์แวร์ที่รองรับ Java
- J2SE 1.4 เปิดตัวพร้อมคุณสมบัติหลักของ Java Web Start, I / O ที่ไม่บล็อก, Logging API, Preferences API และนิพจน์ทั่วไป
- ที่ JavaOne ที่พัฒนา Java ของ Sun เปิดตัว Duke's Choice Awards ประจำปีครั้งแรกซึ่งยกย่องนักประดิษฐ์ที่ใช้เทคโนโลยี Java ในรูปแบบที่ไม่เหมือนใคร
- เว็บไซต์ Java.com เปิดตัวเพื่อให้ผู้บริโภคดาวน์โหลดแพ็คเกจ Java runtime ลงบนเดสก์ท็อปและแล็ปท็อปเพื่อให้สามารถรันแอปพลิเคชัน Java ได้
- NASA แสดงต้นแบบยานสำรวจดาวอังคารบนเวทีที่ JavaOne ซึ่งแสดงให้เห็นว่าเทคโนโลยี Java จะช่วยนำทางจากระยะไกลจาก Mission Control Center ของ NASA ได้อย่างไร
- นักพัฒนามืออาชีพประมาณ 75 เปอร์เซนต์ใช้ Java เป็นภาษาหลักในการพัฒนา
- J2SE 5.0 ถูกนำออกใช้พร้อมคุณสมบัติหลักของ generics คำอธิบายประกอบการแจกแจงและอาร์กิวเมนต์ตัวแปร (varargs)
- การถกเถียงกันว่าจะปล่อย Java เป็นซอฟต์แวร์โอเพนซอร์สเริ่มต้นบนเวทีที่ JavaOne หรือไม่
- สเปิร์ตโรเวอร์ของ NASA ลงจอดบนดาวอังคารเมื่อวันที่ 4 มกราคม พ.ศ.

- ภายในสิ้นปี 2547 Java กำลังทำงานบนอุปกรณ์ 1.5 พันล้านเครื่อง ประยุกต์ใช้งานใน ระบบบันทึกการดูแลสุขภาพอิเล็กทรอนิกส์, telematics, ยานสำรวจดาวอังคาร, ชิมการ์ด

พ.ศ. 2548

- Java ฉลองวันเกิดครบรอบ 10 ปี
- Sun ประเมินว่า Java สร้างรายได้มากกว่า 100 พันล้านเหรียญสหรัฐต่อปี มีนักพัฒนา Java มากกว่า 4.5 ล้านคนอุปกรณ์ที่ใช้ Java 2.5 พันล้านเครื่องและสมาร์ตการ์ดที่ใช้ Java 1 พันล้านเครื่อง
- โปรแกรม Java Champions เปิดตัวที่ JavaOne เพื่อยกย่องผู้นำในชุมชนนักพัฒนา

พ.ศ. 2549–2551

- Sun เผยแพร่ Java สำหรับการพัฒนาโอเพนซอร์สภายใต้ GNU General Public License ซึ่งเป็นใบอนุญาตเดียวกับที่ควบคุมการใช้งานและการพัฒนาระบบปฏิบัติการ Linux
- Java SE 6 เปิดตัวแล้ว
- Java HotSpot VM และคอมไพเลอร์ถูกเผยแพร่ภายใต้ GNU General Public License
- ภาพมาสดอต Duke เปิดตัวภายใต้ลิขสิทธิ์ BSD
- เทคโนโลยี Java ถูกใช้โดยนักพัฒนามากกว่า 6 ล้านคนและทำงานบนอุปกรณ์มากกว่า 5.5 พันล้านเครื่อง
- การประชุมสุดยอดภาษา Java Virtual Machine (JVM) ครั้งแรกจัดขึ้นในเดือนกันยายน 2551 ที่เมืองซานตาคลารา รัฐแคลิฟอร์เนีย เรียนรู้เพิ่มเติม: openjdk.java.net/projects/mlvm/jvmlangsummit/

พ.ศ. 2552–2554

- Oracle ประกาศซื้อกิจการ Sun. Larry Ellison เข้าร่วมเป็นผู้นำ Sun บนเวทีที่ JavaOne ในปี 2009 เพื่อพูดคุยเกี่ยวกับความมุ่งมั่นของ Oracle ในการลงทุนในเทคโนโลยี Java เพื่อประโยชน์ของลูกค้าและชุมชน
- Oracle เปิดตัว Java Magazine ซึ่งเป็นสิ่งพิมพ์ทางเทคนิคที่เขียนโดยนักพัฒนาสำหรับนักพัฒนา ในฐานะที่เป็นสิ่งพิมพ์อย่างเป็นทางการเกี่ยวกับ Java จึงมีโปรไฟล์ของแอป Java ใหม่บทความวิธีใช้ทางเทคนิค ข่าวสารชุมชนและอื่น ๆ
- Java SE 7 เปิดตัวพร้อมคุณสมบัติใหม่ซึ่งรวมถึง Project Coin, invokedynamic, fork / join framework และ API ระบบไฟล์ใหม่ (NIO.2) ประยุกต์ใช้ใน อากาศยานไร้คนขับ:

พ.ศ. 2555–2557

- 97 เปอร์เซนต์ของเดสก์ท็อประดับองค์กรใช้ Java
- Java SE 8 เปิดตัวพร้อมคุณสมบัติหลักของนิพจน์แลมบ์ดาอธิบายประกอบประเภท Java และวันที่และเวลา API
- สิ่งพิมพ์ทางเทคนิค Java SE 8 มากกว่า 80 รายการในแปดภาษามีให้บริการทั่วโลก
- นักพัฒนามากกว่า 9 ล้านคนทั่วโลกใช้ Java ภายในสิ้นปี 2014 ประยุกต์ใช้กิจการ **NASA**

พ.ศ. 2558

- Java ฉลองวันเกิดครบรอบ 20 ปี
- Java เป็นแพลตฟอร์มสำหรับนักพัฒนาอันดับ 1 ของโลก
- นักพัฒนามากกว่า 10 ล้านคนใช้ Java ทั่วโลก
- อุปกรณ์ 13 พันล้านเครื่องรัน Java ประยุกต์ใช้งานกับ Twitter, Netflix

พ.ศ. 2559

- อุปกรณ์ 15 พันล้านเครื่องรัน Java
- Java Magazine มีสมาชิกมากกว่า 250,000 คน
- ประยุกต์ใช้งานกับ Uber, CERN

พ.ศ. 2560

- Java SE 9 เปิดตัวพร้อมคุณสมบัติหลักของ Project Jigsaw (Java Platform Module System), jshell, การคอมไพล์ล่วงหน้า, jlink และสตริงขนาดกะทัดรัด เรียนรู้เพิ่มเติม
- Java ได้รับการจัดอันดับให้เป็นภาษาโปรแกรมอันดับ 1
- นักพัฒนา 12 ล้านคนใช้งาน Java ทั่วโลก
- มี Java Virtual Machines (JVM) ที่ใช้งานอยู่ 38 พันล้านเครื่อง
- มี JVM ที่เชื่อมต่อกับคลาวด์ 21 พันล้านเครื่อง
- เพื่อเร่งการพัฒนาวัตรกรรมของนักพัฒนา Oracle ขอแนะนำช่วงเวลาการเผยแพร่ 6 เดือนสำหรับ Java โดยเริ่มต้นด้วย Java SE 10 ในปี 2018

พ.ศ. 2561

- Java ได้รับการจัดอันดับให้เป็นภาษาอันดับ 1 ที่นักพัฒนาใช้สำหรับระบบคลาวด์
- โปรแกรม Java Champions มีสมาชิก 150 คน
- Java Community Process ฉลองวันเกิดครบรอบ 20 ปี
- Oracle ประกาศว่าจะเปิดคุณสมบัติ Java เชิงพาณิชย์หลายอย่างรวมถึงการแชร์ข้อมูลคลาสของแอปพลิเคชัน, Z Garbage Collector (ZGC), Oracle Java Flight Recorder และ Oracle Java Mission Control
- Java SE 10 เป็นรุ่นแรกในรอบระยะเวลาการเผยแพร่หกเดือน คุณลักษณะใหม่ ได้แก่ การอนุมานประเภทตัวแปรโลคัลการแชร์ข้อมูลคลาสของแอ็พพลิเคชันการกำหนดเวอร์ชันรีลีสตามเวลาตัวรวบรวมขยะ G1 แบบขนานเต็มใบรับรองรูปและการจับมือเทรดโลคัล เรียนรู้เพิ่มเติม
- Oracle เพิ่มตัวเลือกการออกไปอนุญาตเชิงพาณิชย์ของ Java สำหรับนักพัฒนาและ บริษัท ที่ต้องการการสนับสนุนระดับองค์กรด้วยการแนะนำโปรแกรมการสมัครสมาชิก Java SE
- Java SE 11 เปิดตัว; คุณลักษณะใหม่ ได้แก่ โคลเอ็นต์ HTTP, Oracle Java Flight Recorder, เปิดโปรแกรมซอร์สโค้ดไฟล์เดียว, Transport Layer Security (TLS) 1.3 และ ZGC ซึ่งเป็นตัวรวบรวมขยะที่มีเวลาแฝงต่ำที่ปรับขนาดได้และทดลอง

พ.ศ. 2562

- Java ได้รับการจัดอันดับให้เป็นภาษาอันดับ 1 ที่นักพัฒนาใช้สำหรับระบบคลาวด์อีกครั้ง
- Java SE 12 เปิดตัวพร้อมคุณสมบัติหลักของนิพจน์สวิตช์ (ตัวอย่างแรก), JVM Constants API และไฟล์เก็บถาวรการแชร์ข้อมูลคลาสเริ่มต้น (CDS) เรียนรู้เพิ่มเติม
- Java SE 13 เปิดตัวพร้อมคุณสมบัติหลักของไฟล์เก็บถาวร CDS แบบไดนามิกความสามารถในการยกเลิกการคอมมิตหน่วยความจำที่ไม่ได้ใช้การสลับนิพจน์ (การแสดงผลตัวอย่างครั้งที่สอง) และบล็อกข้อความ (การแสดงผลตัวอย่างครั้งแรก) เรียนรู้เพิ่มเติม ประยุกต์ใช้ทำงานกับ Spotify, Siemens, Ocado, Satellite Applications

พ.ศ. 2563

- Java SE 14 เปิดตัวพร้อมคุณสมบัติหลักของการจับคู่รูปแบบสำหรับอินสแตนซ์ของ (การแสดงผลตัวอย่าง), การสตรีมเหตุการณ์ JDK Flight Recorder, บัฟเฟอร์ไบต์ที่แมปแบบไม่ลบเลือน, ข้อยกเว้นของพอยน์เตอร์ว่างที่เป็นประโยชน์, เรกคอร์ด (แสดงผลตัวอย่าง), การสลับนิพจน์, บล็อกข้อความ (การแสดงผลตัวอย่างครั้งที่สอง), ZGC สำหรับ macOS และ Windows, API การเข้าถึงหน่วยความจำต่างประเทศ (ศูนย์บ่มเพาะ) และเครื่องมือบรรจุกัณฑ์ (ต้อบ) เรียนรู้เพิ่มเติม
- Java ฉลองวันเกิดครบรอบ 25 ปี
- Java SE 15 เปิดตัวพร้อมคุณสมบัติหลักของคลาสปิดผนึก (ตัวอย่าง) คลาสที่ซ่อนอยู่การจับคู่รูปแบบสำหรับอินสแตนซ์ของ (การแสดงผลตัวอย่างครั้งที่สอง) บล็อกข้อความและบันทึก (การแสดงผลตัวอย่างครั้งที่สอง) เรียนรู้เพิ่มเติม
- Java ยังคงเป็นภาษาโปรแกรมอันดับ 1 สำหรับนักพัฒนา
- Oracle เปิดตัว Inside Java นำเสนอข่าวสารและอัปเดตจากทีม Java ของ Oracle เรียนรู้เพิ่มเติม
- Oracle เปิดตัวซีรีส์ Inside Java podcast เรียนรู้เพิ่มเติม ประยุกต์กับ Genetics [https://genetics.io_United States Geological Survey](https://genetics.io_United_States_Geological_Survey) , NASA Open Code, Minecraft, MATLAB , Robot Java + STEM

การติดตั้งโปรแกรมภาษาจาวาและเครื่องมือพัฒนาโปรแกรม

การพัฒนาหัตถ์ฉบับของภาษาจาวา

แนวทางการพัฒนาแบ่งออกเป็น 2 แนวทางดังนี้

1. ใช้โปรแกรมประเภทอิดิตเตอร์ เป็นเครื่องมือในการเขียน โปรแกรมกลุ่มนี้มีขีดความสามารถไม่สูงนัก แนวทางนี้จะใช้อิดิตเตอร์ในการเขียนรหัสต้นฉบับและรันโปรแกรมผ่านอิดิตเตอร์หรือรันผ่านคอมมานด์ไลน์ (Command Line) ของระบบปฏิบัติการ หากเครื่องมือใดสามารถเชื่อมกับเจเอสไอได้โดยตรงก็สามารถรันผ่านอิดิตเตอร์ได้เลย หากโปรแกรมนั้นไม่สามารถติดต่อกับเจเอสไอได้ต้องรันผ่านคอมมานด์ไลน์ของระบบปฏิบัติการเท่านั้น สิ่งที่สำคัญ คือ ต้องเขียนรหัสต้นฉบับเองทั้งหมด หลังจากนั้นใช้คำสั่ง `javac` เพื่อคอมไพล์โปรแกรม แล้วรันโดยใช้คำสั่ง `java` เป็นต้น

การใช้งานแบบนี้ต้องติดตั้งโปรแกรมภาษาจาวาลงในเครื่องก่อน หลังจากนั้นกำหนดพาท (Path) และ คลาสพาท (Class path) ให้ขึ้นกับสภาวะแวดล้อมของระบบปฏิบัติการแต่ละประเภทเพื่อที่จะเรียกคอมไพเลอร์ (javac.exe) ได้ถูกต้อง โปรแกรมอิดิตเตอร์ที่ใช้ในการพัฒนารหัสต้นฉบับได้ เช่น Notepad, EditPlus, jCreator และโปรแกรมอิดิตเตอร์อื่น ๆ ที่สามารถบันทึกเป็นแฟ้มให้มีนามสกุลเป็น .java ได้

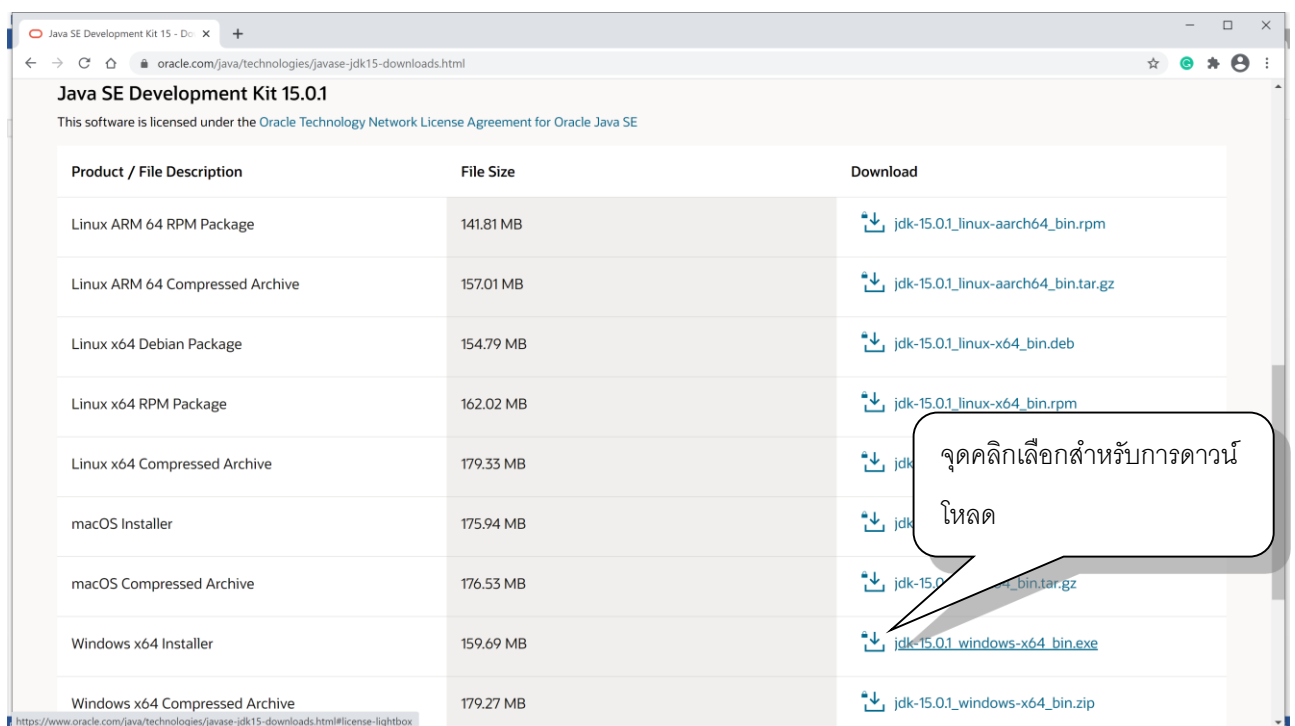
ข้อดีของการพัฒนาแบบนี้ คือ ผู้เขียนโปรแกรมไม่ต้องใช้เครื่องที่ประสิทธิภาพสูงนักก็สามารถเขียนโปรแกรมเพื่อใช้งานได้ แต่ข้อเสีย คือ ถ้าพัฒนาแอปพลิเคชัน (Application) ขนาดใหญ่จะทำได้ช้าเพราะต้องเขียนรหัสต้นฉบับจำนวนมาก

2. พัฒนาโดยใช้จิวไอทูล (GUI tools) ซึ่งเป็นเครื่องมือที่ใช้งานในแนวภาษายุคที่ 4 แนวทางการพัฒนา คือ การคลิกคอมโพเนนต์ (Component) แล้วดรอปลงบนฟอร์ม แล้วเขียนรหัสต้นฉบับเหมือนภาษาชั้นสูงที่เป็นค่ายปิดทั่วไป เครื่องมือเหล่านี้มีทั้งให้ฟรีและต้องซื้อ ตัวอย่างโปรแกรมสำหรับแจกฟรี เช่น NetBeans, Sun studio, Eclipse และโปรแกรมสำหรับขาย เช่น Visual Paradigm, และ JBuilder เป็นต้น

ข้อดีของแนวการพัฒนาแบบนี้ คือ สามารถทำงานได้เร็วเหมาะแก่การพัฒนาโปรแกรมที่มีขนาดใหญ่ แต่ข้อเสีย คือ ต้องใช้เครื่องคอมพิวเตอร์ที่ประสิทธิภาพสูง ๆ โดยเฉพาะแรมควรมีขนาดตั้งแต่ 2 กิกะไบต์ขึ้นไป

การดาวน์โหลด

แหล่งดาวน์โหลดโปรแกรมที่เกี่ยวข้องกับการใช้งานภาษาจาวาอยู่ที่เว็บไซต์ <https://www.oracle.com/java/technologies/javase-jdk15-downloads.html> ขั้นตอนของการดาวน์โหลด และคอมไพล์เลอร์ของภาษาจาวา หรือ <https://www.oracle.com/java/technologies/javase-downloads.html> เมื่อเข้าไปที่เว็บไซต์ดังกล่าว แล้วเลือก ในส่วนของ Software Downloads ดังภาพที่ 2.2 แล้วคลิกเลือกที่ Java SE



ภาพที่ 1.5 การเข้าไปดาวน์โหลดโปรแกรมภาษาจาวา

หรืออีกแนวทาง คือ เป็น jdk.java.net ซึ่งเป็นโอเพนซอร์สจาวา



ภาพที่ 1.6 การเข้าไปดาวน์โหลดโปรแกรมภาษาจาวา

เมื่อดาวน์โหลดมาแล้ว จะได้แฟ้มดังนี้

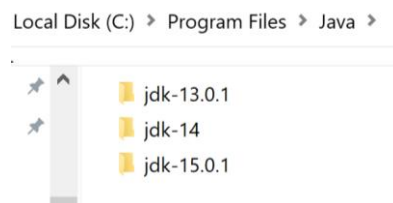
 jdk-15.0.1_windows-x64_bin	22/12/2563 13:26	Application	163,528 KB
 openjdk-15.0.1_windows-x64_bin	22/12/2563 13:25	WinRAR ZIP archive	191,345 KB

การติดตั้งภาษาจาวา

หลังจากดาวน์โหลดโปรแกรมมาเรียบร้อยแล้ว ขั้นตอนต่อไปต้องติดตั้งโปรแกรมลงเครื่องเหมือนกับโปรแกรมทั่วไป หลังจากนั้นต้องกำหนดเส้นทางให้กับการเรียกใช้คำสั่งสำหรับการคอมไพล์และรันโปรแกรม

ดับเบิลคลิก  jdk-15.0.1_windows-x64_bin จากนั้นติดตั้งตามลำดับของโปรแกรม

เมื่อติดตั้งสำเร็จแล้ว จะได้ตำแหน่งของการติดตั้งดังนี้



ภาพที่ 1.7 แสดงตำแหน่งจัดเก็บโปรแกรมภาษาจาวาหลังการติดตั้ง

โดยที่ภายในจะบรรจุ โปรแกรมและไฟล์เดอร์ ดังนี้



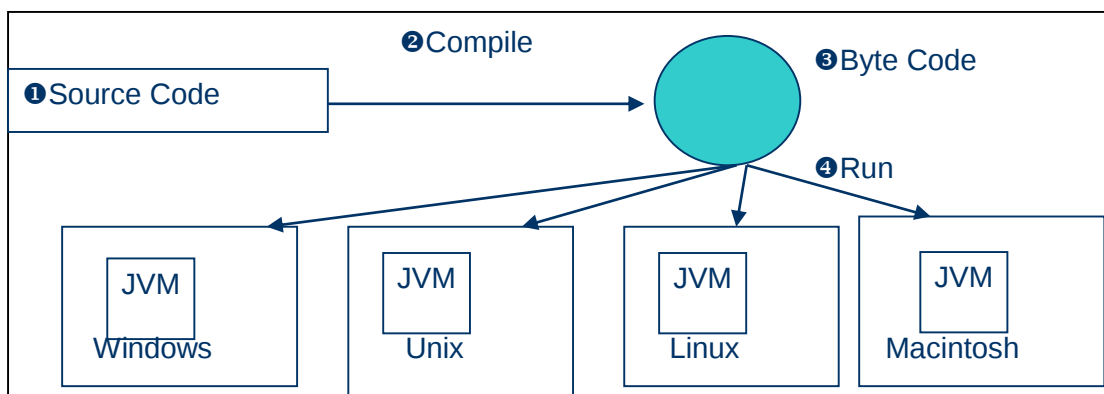
ภาพที่ 1.8 แสดงโฟลเดอร์ส่วนประกอบของภาษาจาวาหลังการติดตั้ง

โครงสร้างและรูปแบบพื้นฐานของภาษาจาวา

“Write Once Run Anywhere” เป็นคำขวัญของจาวา ซึ่งหมายถึงโปรแกรมภาษาจาวานั้นเขียนเพียงครั้งเดียวแต่สามารถรันได้ทุก ๆ ที่หรือทุก ๆ แพลตฟอร์ม การนำเสนอในตำราเล่มนี้จะนำเสนอการเขียนจาวา แอปพลิเคชันเป็นหลัก ขั้นตอนการเขียน ผู้เขียนโปรแกรมจะต้องเขียนรหัสต้นฉบับ และบันทึกแฟ้มลงในดิสก์ หลังจากนั้นคอมไพล์โปรแกรมจะได้ไบนารีรหัสต้นฉบับเป็นแฟ้มนามสกุล .class หลังจากนั้นจึงรันโปรแกรม โดยการรันแต่ละแพลตฟอร์มจะต้องมี JVM (Java Virtual Machine) ที่เขียนขึ้นมาเฉพาะรูปแบบนั้น ๆ เพื่อจัดสภาวะการรันในแต่ละระบบปฏิบัติการ

1. จาวาแอปพลิเคชัน

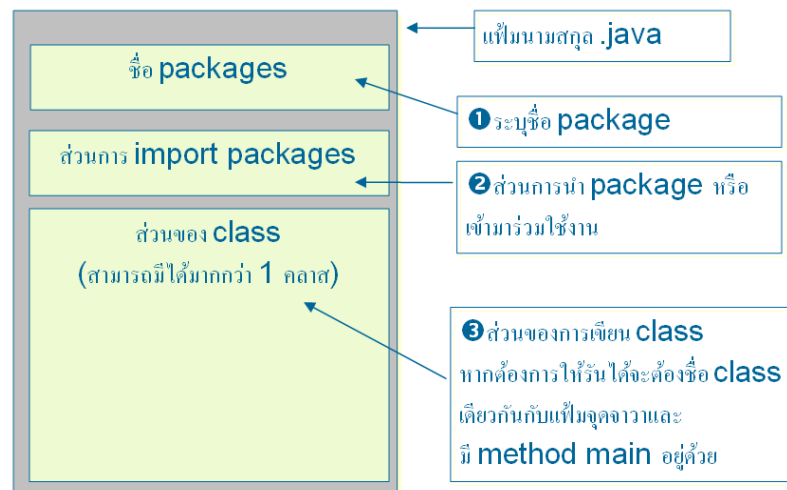
ขั้นตอนการเขียนแบบนี้ ผู้เขียนโปรแกรมจะต้องเขียนรหัสต้นฉบับ หลังจากนั้นคอมไพล์ให้เป็นไบนารีโค้ดเป็นแฟ้มนามสกุล .class ที่สามารถนำไปรันได้หลายแพลตฟอร์ม โดยการรันแต่ละแพลตฟอร์มจะต้องมีจาวาเวอร์ชวลแมชีนของระบบนั้น ๆ เพื่อจัดสภาวะการรันในแต่ละระบบปฏิบัติการ ภาพที่ 2.15 แสดงรายละเอียดดังกล่าว



ภาพที่ 2.15 แสดงหลักการทำงานของภาษาจาวา

2. โครงสร้างของโปรแกรม

โครงสร้างของโปรแกรมภาษาจาวามีความยืดหยุ่นมากกว่าภาษาในยุคที่สาม สามารถสร้างโปรแกรมเพียงหนึ่งโปรแกรมหรือมากกว่าหนึ่งโปรแกรมด้วยแฟ้มเดียวกัน โดยระบุชื่อของแฟ้มให้มีนามสกุล .java และมีคลาสหลาย ๆ คลาส โครงสร้างของรหัสต้นฉบับแสดงดังภาพที่ 1.9



ภาพที่ 1.9 แสดงโครงสร้างรหัสต้นฉบับของภาษาจาวา

2.1 ส่วน ❶ เป็นส่วนของการระบุชื่อแพ็คเกจ ที่อาจจะระบุหรือไม่ก็ได้ สำหรับการเขียนโปรแกรมที่ไม่มี ความซับซ้อนมาก แต่หากเป็นโปรแกรมที่มีความซับซ้อน จำเป็นต้องสร้างที่เก็บหลายที่ ดังนั้นจำเป็นต้อง กำหนดชื่อแพ็คเกจ เพราะการเรียกใช้จะต้อง import โปรแกรมที่เกี่ยวข้องเข้ามาใช้งานร่วมด้วย

2.2 ส่วน ❷ เป็นส่วน import package เป็นส่วนการนำเอาแพ็คเกจที่เก็บคำสั่งที่เกี่ยวข้องกับส่วน ของโปรแกรมที่จะเขียนเข้ามารวมเพื่อให้ใช้งานคำสั่งได้ เช่น java.io.* เป็นแพ็คเกจสำหรับการเรียกใช้งาน คำสั่งเกี่ยวกับการติดต่อกับอุปกรณ์ javax.swing.* เป็นแพ็คเกจสำหรับการเขียนโปรแกรมแบบกราฟิก เป็นต้น

2.3 ส่วน ❸ เป็นส่วนของตัวคลาสและเมธอด ส่วนนี้ต้องเขียนคลาสและส่วนเมธอดที่เป็นฟังก์ชันการ ทำงานย่อย ๆ เพื่อใช้งานโดยอาจมีเพียงคลาสเดียว หรือหลายคลาสก็ได้ แสดงตัวอย่างของโครงโปรแกรม ทั้ง 2 แบบ ดังนี้

ตัวอย่างโครงโปรแกรม แบบที่ 1

```
import java.io.*; //ส่วนการ import
public class firstClass //ชื่อคลาส
{
    public static void main(String args[]) //ส่วนของเมธอดหลัก
    {
        // เนื้อหาของโปรแกรม
    }
}
```

โครงสร้างโปรแกรม แบบที่ 2

```
import java.io.*; //ส่วนการ import
```

```

public class firstClass //ชื่อคลาสแรก
{
    public static void main(String args[]) //ส่วนของเมธอดหลัก
    {
        // เนื้อหาของโปรแกรมหลัก
    }
}

class secondClass // ชื่อคลาสที่สอง
{
    public void metA(void)
    {
        // เนื้อหาของโปรแกรม method A
    }
}

```

ตัวอย่างที่ 1.1 จงเขียนโปรแกรมแสดงข้อความ Hello world บนจอภาพ

รหัสต้นฉบับ :

```

import java.io.*; //ส่วนการ import
public class firstClass //ชื่อคลาส
{ //เริ่มคลาส
    public static void main(String args[]) //ส่วนเมธอดหลัก
    { //เริ่มต้นคำสั่งของเมธอด main
        System.out.println("Hello world"); //แสดงข้อความตามที่โจทย์ต้องการ
    } //สิ้นสุดคำสั่งของเมธอด main
} //สิ้นสุดคลาส

```

ผลการรันของโปรแกรมนี้อาจจะแสดงข้อความ Hello world ตามที่โจทย์ต้องการ

3. ไวยากรณ์ (Syntax)

การเขียนโปรแกรมด้วยภาษาใดก็ตาม การเรียนรู้ไวยากรณ์ของโปรแกรมภาษานั้นจำเป็นต้องได้เขียนโปรแกรมให้ทำงานตามที่ต้องการได้อย่างถูกต้อง และให้ส่วนการคอมไพล์ได้แปลคำสั่งที่ถูกต้องเพื่อส่งคอมพิวเตอร์ รูปแบบของโปรแกรมที่จะเขียนมีรูปแบบโดยรวมดังภาพที่ 2.16 โดยมีส่วนของการตั้งชื่อคลาส ส่วนการ import และส่วนของเขียนเมธอดต่าง ๆ ในส่วนนี้จะอธิบายโดยภาพรวมได้ดังนี้

1) การตั้งชื่อโปรแกรม .java หรือชื่อตัวแปรใด ๆ ก็ตาม ห้ามตั้งชื่อซ้ำกับชื่อสงวน (Reserve Words) ของภาษาจาวา มีดังนี้

abstract	else	Interface	super
boolean	extends	long	switch
break	false	native	synchronized
byte	final	new	this
case	finally	null	throw
catch	float	package	throws
char	for	private	transient
class	if	protected	true
continue	implements	public	try
default	import	return	void
do	instanceof	short	volatile
double	int	static	while

ภาพที่ 1.10 แสดงคำสงวนของโปรแกรมภาษา

2) แพคเกจ หมายถึง ส่วนของการเก็บคำสั่ง คลาส สำหรับใช้งานแต่ละอย่างเช่น java.text จะเป็นแพคเกจที่ใช้สำหรับการจัดการเท็กซ์ หรือ java.io เป็นแพคเกจที่รวบรวมคำสั่งและเมธอดการใช้งานของส่วนนำเข้าและแสดงผลเป็นต้น การที่จะ import แพคเกจใด ๆ นั้นจะต้องเรียนรู้ว่าคำสั่งที่ต้องการใช้นั้นอยู่ในแพคเกจใด

3) สัญลักษณ์และเครื่องหมายสำหรับการเขียนโปรแกรม มีดังนี้

{ ใช้สำหรับเริ่มต้นของคลาสและเมธอดต่าง ๆ หรือกลุ่มของคำสั่ง

} ใช้สำหรับสิ้นสุดของคลาสและเมธอดต่าง ๆ หรือกลุ่มของคำสั่ง

; ใช้สำหรับจบคำสั่งแต่ละคำสั่ง หรือการกำหนดค่าให้กับตัวแปรต่าง ๆ

() ใช้สำหรับแสดงส่วนการเรียกใช้เมธอด และคำสั่ง บางครั้งต้องใช้การผ่านค่าพารามิเตอร์ด้วย

/* */ ใช้สำหรับสร้างคำอธิบายเป็นช่วงระหว่างเครื่องหมายทั้งสองส่วนนี้จะไม่ถูกนำไป

คอมไพล์

// ใช้สำหรับอธิบายข้อความเพียงบรรทัดเดียว

@ ใช้สำหรับเพื่อสร้างคำอธิบายประกอบส่วนของคำอธิบายคลาสสำหรับสร้างส่วนของเอกสารประกอบของคลาสหรือเมธอดที่สร้างขึ้น

[] ใช้สำหรับอ้างอิงเพื่อจองหรืออ้างถึงโครงสร้างข้อมูลแบบอาร์เรย์

ตัวอย่างที่ 1.2 จงบอกว่าจุดใดในโปรแกรมไม่ถูกต้องตามหลักไวยากรณ์การเขียนโปรแกรม

รหัสต้นฉบับ :

```
1. import java.io.*
2. public class firstClass
3. (
```

```

4.     public static void main(String args[])
5.     {
6.         System.out.println("Hello world");
7.     }
8. )

```

คำตอบ : บรรทัดที่ผิดคือ

บรรทัดที่ 1 ผิด เพราะไม่มี ; หลังจาก java.io.*

บรรทัดที่ 3 ผิด เพราะ ใช้ (แทนที่จะใช้ {

บรรทัดที่ 8 ผิด เพราะใช้) แทนที่จะใช้ }

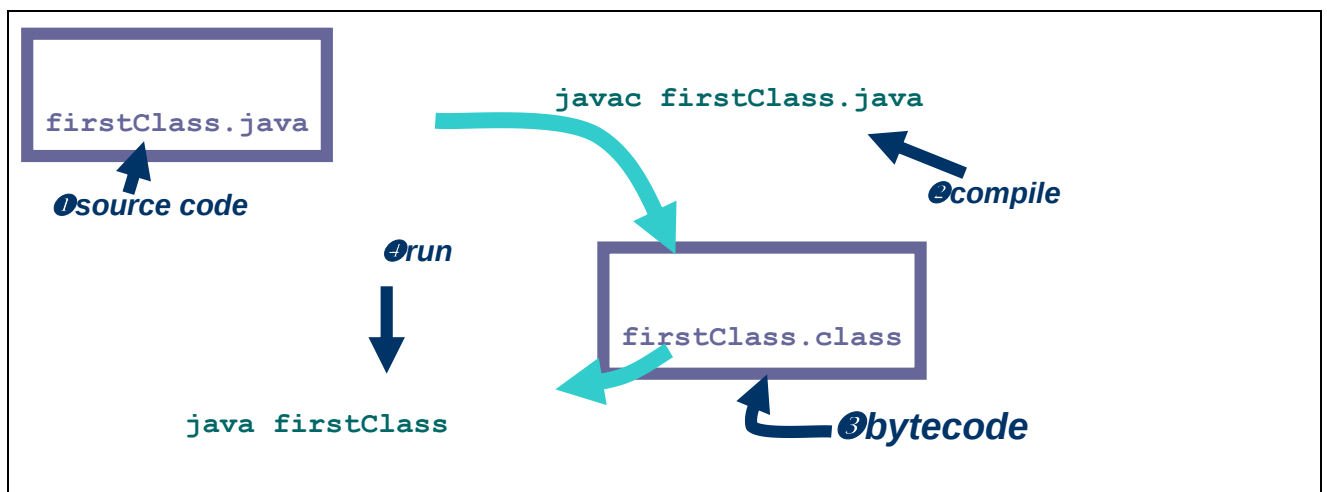
4. การคอมไพล์และรันโปรแกรม

ขั้นตอนคอมไพล์ พิมพ์ javac ตามด้วยชื่อโปรแกรม .java เช่น javac firstClass.java หมายถึงการคอมไพล์แฟ้ม firstClass.java ให้เป็นแฟ้มนามสกุล .class ซึ่งเป็นไบนารีโค้ดนั่นเอง

การรันโปรแกรม พิมพ์ java ตามด้วยชื่อคลาสที่ผ่านการคอมไพล์แล้ว เช่น

java firstClass หมายถึง การรันโปรแกรม firstClass.class

หลักการคอมไพล์และรันโปรแกรมแสดงดังภาพที่ 1.11



ภาพที่ 1.11 แสดงผลการรันและคอมไพล์โปรแกรม

ตัวอย่างที่ 1.3 กำหนดให้ มีแฟ้มภาษาจาวา ดังนี้ jj.java, xx.class, hero.java จงบอกวิธีการที่จะทำให้โปรแกรมดังกล่าวทำงานได้

คำตอบ :

โปรแกรม jj.java และ hero.java จะต้องคอมไพล์ก่อนด้วยคำสั่ง javac jj.java และ javac hero.java หรือหากแฟ้มดังกล่าวเก็บไว้ที่เดียวกันอาจใช้คำสั่ง javac *.java ก็ได้ และหลังจากนั้นทำการรันโปรแกรมโดย

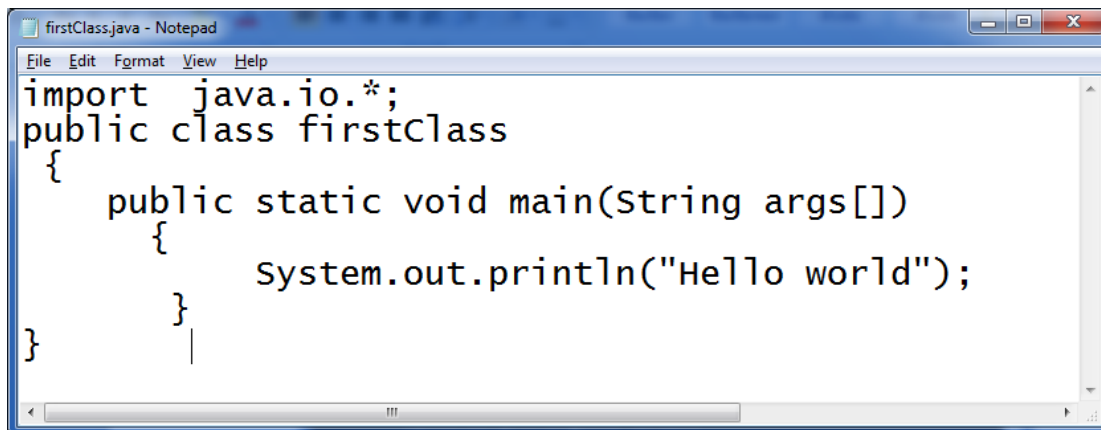
1) java jj

2) java hero และ

3) java xx ตามลำดับ

ตัวอย่างที่ 1.4 จากตัวอย่างที่ 1.2 จงแสดงแก้ไขข้อผิดพลาดของโปรแกรมและแสดงวิธีการเขียนโปรแกรมด้วยโปรแกรมโน้ตแพทและเครื่องมือช่วยพัฒนาแบบสภาวะแวดล้อมแบบกราฟิก และรันโปรแกรม

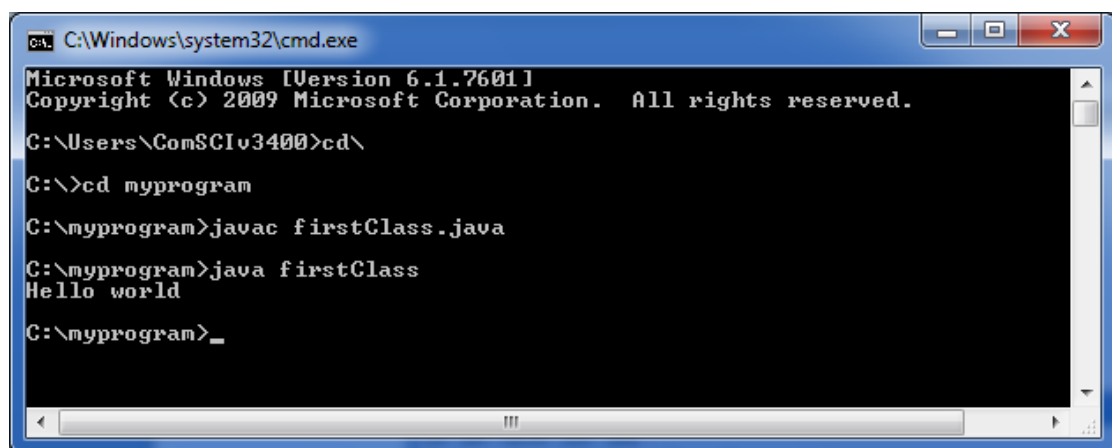
จากตัวอย่างดังกล่าว ภาพที่ 1.11 แสดงตัวอย่างการเขียนโปรแกรมด้วยโปรแกรมโน้ตแพท ภาพที่ 1.12 แสดงการคอมไพล์และการรันโดยผ่านคำสั่งดอสโดยตรง สำหรับภาพที่ 1.13 แสดงการเขียนโปรแกรมและผลการรันเมื่อใช้โปรแกรมเน็ตบีน ช่วยในการพัฒนาโปรแกรม



```
firstClass.java - Notepad
File Edit Format View Help
import java.io.*;
public class firstClass
{
    public static void main(String args[])
    {
        System.out.println("Hello world");
    }
}
```

ภาพที่ 1.12 แสดงการเขียนโปรแกรมด้วยโปรแกรมโน้ตแพท

จากภาพที่ 1.12 โปรแกรมเก็บไว้ที่ c:\myprogram และตั้งชื่อโปรแกรมให้ตรงกับชื่อของคลาสคือ firstClass.java สำหรับภาพที่ 1.13 แสดงการเข้าไปยังโฟลเดอร์ c:\myprogram ด้วยคำสั่ง cd\ และ cd myprogram หลังจากนั้นคอมไพล์โปรแกรมด้วยคำสั่ง javac firstClass.java และรันโปรแกรมด้วย java firstClass ซึ่งจะได้ผลการรันโดยแสดงข้อความว่า Hello World



```
C:\Windows\system32\cmd.exe
Microsoft Windows [Version 6.1.7601]
Copyright (c) 2009 Microsoft Corporation. All rights reserved.

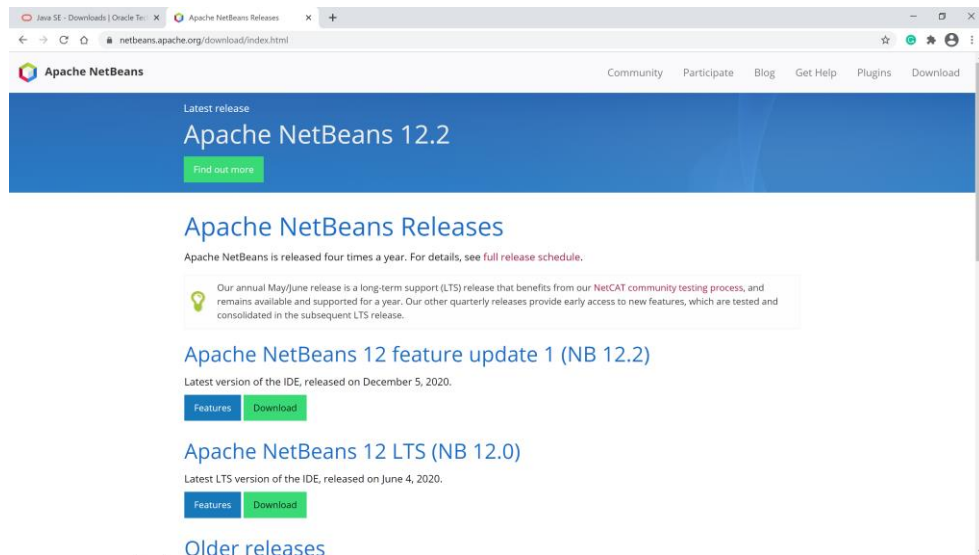
C:\Users\ComSCIv3400>cd\
C:\>cd myprogram
C:\myprogram>javac firstClass.java
C:\myprogram>java firstClass
Hello world
C:\myprogram>_
```

ภาพที่ 1.13 แสดงผลการคอมไพล์และรันโปรแกรม

การพัฒนาโปรแกรมภาษาจาวาด้วยเน็ตบินไอดี

สำหรับการพัฒนาโปรแกรมด้วยเครื่องมือช่วยอย่างอื่น เช่นโปรแกรมเน็ตบิน ผู้พัฒนาจะต้องดาวน์โหลดโปรแกรมดังกล่าวมาก่อนและติดตั้งให้สำเร็จหลังจากนั้นจึงจะสามารถพัฒนาโปรแกรม ซึ่งโปรแกรมเน็ตบินจะให้สร้างโปรเจกต์ขึ้นมาก่อนเป็นอันดับแรก หลังจากนั้นจึงพัฒนาโปรแกรม

การดาวน์โหลดได้ที่ www.netbeans.apache.org/download/index.html ดังนี้

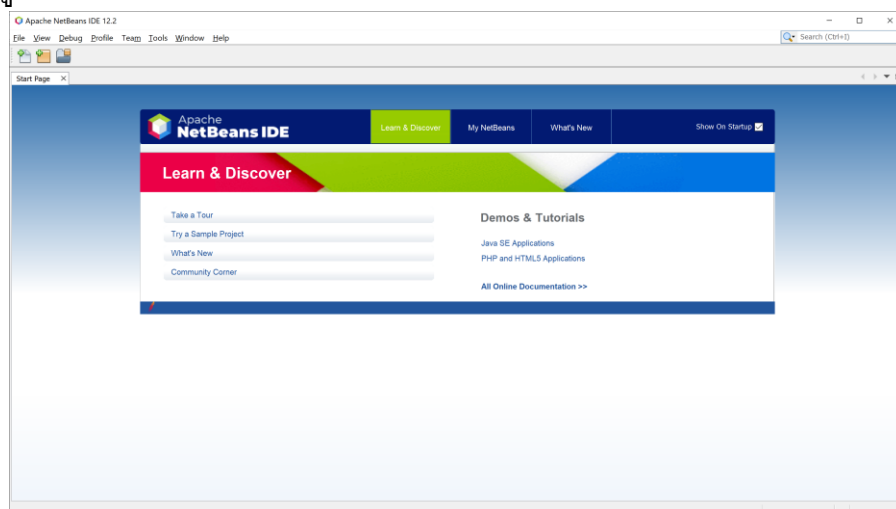


ภาพที่ 1.14 แสดงตำแหน่งการดาวน์โหลดโปรแกรมเน็ตบิน

จากนั้นติดตั้งโปรแกรมเน็ตบิน จาก [Apache-NetBeans-12.2-bin-windows-x64](#) ซึ่งการติดตั้งจะต้องระบุเส้นทางที่ติดตั้งโปรแกรมภาษาจาวาให้กับโปรแกรมเน็ตบินด้วย หลังจากติดตั้งเรียบร้อยแล้วจะมีไอคอนเส้นทางลัดเข้าโปรแกรม ดังนี้

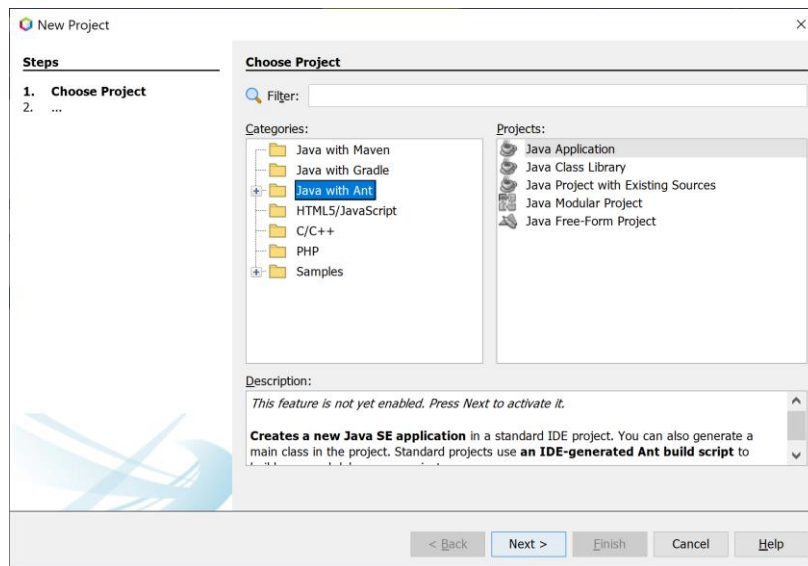


เมื่อเข้าสู่โปรแกรม จะได้โปรแกรมที่มีลักษณะดังนี้



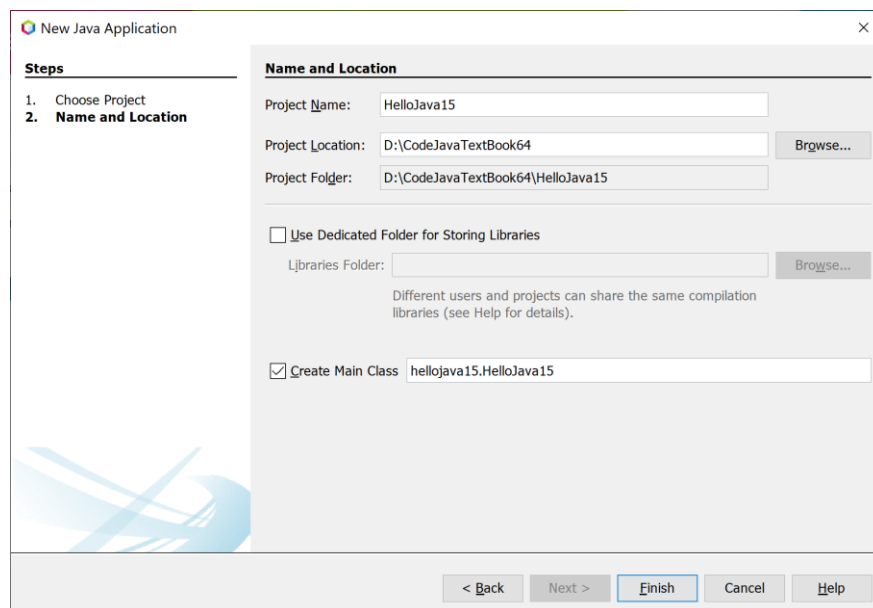
ภาพที่ 1.15 แสดงการเริ่มใช้งานโปรแกรมเน็ตบิน

การเริ่มเขียนโปรแกรม เลือกเมนู **Files->New Project** แล้วเลือก **Java with Ant -> Java Application** ดังภาพ



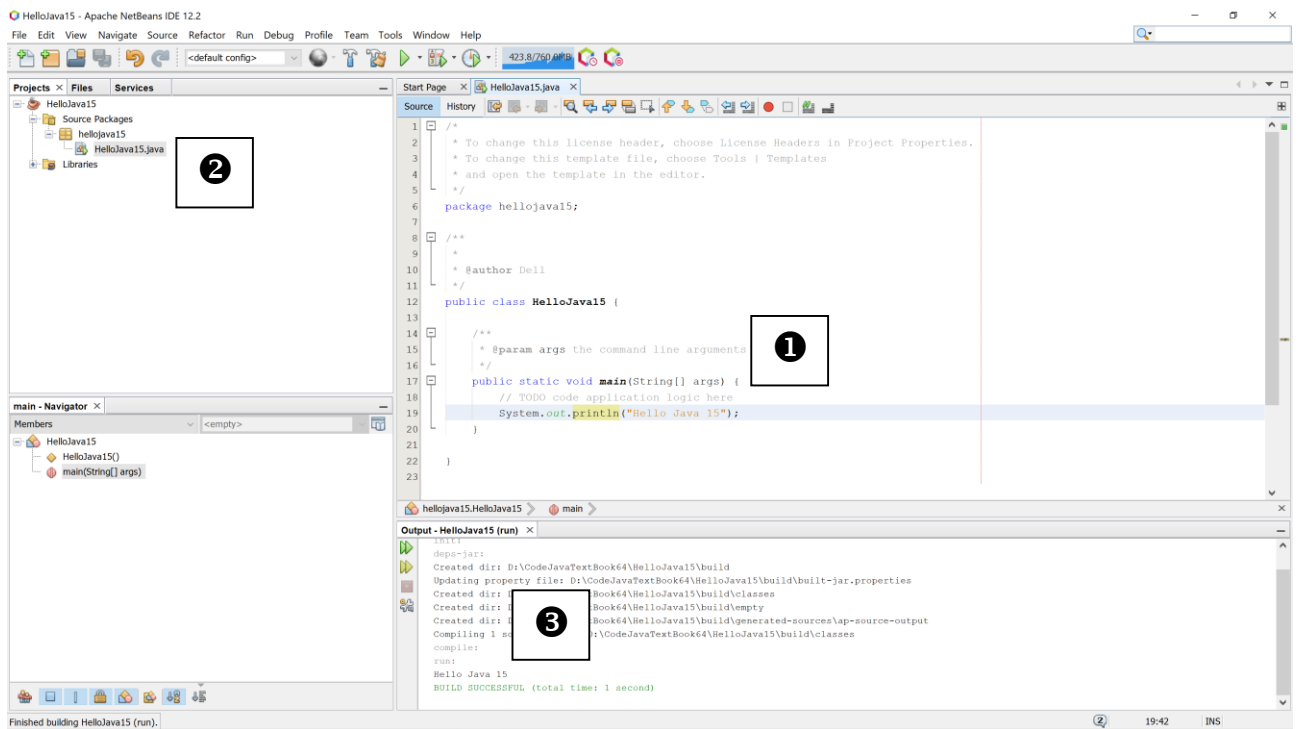
ภาพที่ 1.16 เริ่มต้นการเขียนโปรแกรมด้วยโปรแกรมเนตเบิน

จากนั้นกำหนดชื่อโปรเจกต์ และตำแหน่งจัดเก็บโปรแกรม ตัวอย่างดังภาพ



ภาพที่ 1.17 กำหนดโปรเจกต์และตำแหน่งจัดเก็บโปรแกรม

จากนั้นสามารถเขียนโปรแกรมและรันโปรแกรม โดยคลิกเมาส์ขวา เลือก run file แสดงตัวอย่างการเขียนโปรแกรมและผลการรันดังภาพ แสดงตัวอย่างการพัฒนาโปรแกรม ภาพที่ 1.18 ❶ เป็นส่วนของชื่อแฟ้ม HelloJava15.java ❷ และ ❸ เป็นส่วนของผลการรันโปรแกรม



ภาพที่ 1.18 แสดงการพัฒนา คอมไพล์และผลการรันโปรแกรมเมื่อใช้โปรแกรมเนตเบิน

ในการพัฒนาโปรแกรมในบทต่อๆ จะไม่ได้แสดงภาพประกอบด้วยโปรแกรมเนตเบิน แต่จะแสดงเฉพาะรหัสโปรแกรมและผลการรันเท่านั้น ยกเว้นในบทที่ 10 ซึ่งแสดงการพัฒนาโปรแกรมภาษาจาวาแบบกราฟิก อินเทอร์เฟซ ซึ่งจะแสดงส่วนของคอมไพล์เนตการกำหนดคุณลักษณะ และการเขียนโปรแกรมให้เห็นต่อไป

สรุป

โปรแกรมภาษาจาวาเป็นโปรแกรมภาษาได้รับความนิยมทั้งในอดีตและในปัจจุบัน เพราะเป็นซอฟต์แวร์แบบโอเพ่นซอร์สที่สามารถดาวน์โหลดมาใช้งานได้โดยไม่มีค่าใช้จ่าย การพัฒนาโปรแกรมด้วยภาษาจาวา สามารถพัฒนาจากเครื่องมือที่เป็นโปรแกรมประเภทอิดีเตอร์ทั่วไป และพัฒนาด้วยเครื่องมือที่เป็นไอดีอีทูล ที่สามารถพัฒนาโปรแกรมได้รวดเร็ว ตามแนวทางการพัฒนาแบบภาษายุคที่ 4 ที่นิยมใช้เขียนโปรแกรมในปัจจุบัน หลักการเขียนรหัสต้นฉบับและรูปแบบไวยากรณ์ของภาษามีความคล้ายคลึงกับภาษาซีเป็นอย่างมาก และสามารถพัฒนาโปรแกรมเชิงวัตถุได้อย่างเต็มรูปแบบ โครงสร้างของการเขียนโปรแกรมจะเขียนโปรแกรมโดยใช้คลาส ซึ่งคลาสที่จะให้รันโปรแกรมได้นั้น ต้องตั้งชื่อให้ตรงกับชื่อแฟ้มที่บันทึก และต้องมีเมธอด main อยู่ด้วยเสมอ การคอมไพล์โปรแกรมใช้ javac ตามด้วยชื่อแฟ้มนามสกุล .java และรันโปรแกรมโดยใช้ java ตามด้วยชื่อคลาสหลังการคอมไพล์

แบบฝึกหัดท้ายบท

ตอนที่ 1 จงเขียนเครื่องหมาย ✓ หน้าข้อที่กล่าวถูกต้อง และ เขียนเครื่องหมาย ✗ หน้าข้อที่กล่าวผิด

1. _____ ภาษาจาวาเป็นภาษาคอมพิวเตอร์แบบโครงสร้าง
2. _____ ภาษาจาวาทุกโปรแกรมต้องมี main
3. _____ ภาษาจาวาเกิดจากความร่วมมือของบริษัท Sun Micro System กับ IBM
4. _____ จาวา Applet เป็นภาษาจาวาที่เน้นการใช้งานบน Embedded
5. _____ ภาษาจาวาสามารถเขียนเป็นแบบเว็บเซอร์วิสได้
6. _____ เงื่อนไขของการคอมไพล์โปรแกรมต้องมีเมธอด main และชื่อแฟ้มนามสกุล .java
7. _____ ภาษาจาวาเป็นภาษาของนักพัฒนาโปรแกรมที่ต้องมีพื้นฐานภาษาซีเท่านั้น
8. _____ System.exit() ใช้สำหรับสั่งให้ โปรแกรมจบการทำงาน
9. _____ การรันโปรแกรมภาษาจาวา ใช้ javac ตามด้วยชื่อแฟ้ม .java
10. _____ NetBeans, Eclipse, JBuilder ต่างเป็นเครื่องมือเขียนโปรแกรมด้วยจาวา

ตอนที่ 2 จงตอบคำถามต่อไปนี้

1. Write Once Run Anywhere ในมุมมองของภาษาจาวาหมายความว่าอย่างไร
2. จงบอกวิธีการดาวน์โหลดโปรแกรมภาษาจาวา
3. จงอธิบายวิธีการติดตั้งโปรแกรมภาษาจาวา
4. จงอธิบายวิธีการคอมไพล์และรันโปรแกรมด้วยภาษาจาวา
5. จงระบุชื่อของโปรแกรมที่ช่วยในการพัฒนารหัสต้นฉบับของภาษาจาวาอย่างน้อยอย่างน้อย 5 โปรแกรม
6. จงระบุแพลตฟอร์มของภาษาจาวาอย่างน้อย 4 แพลตฟอร์ม
7. จงยกตัวอย่างโครงสร้างโปรแกรมในการคำนวณเพื่อแสดงสูตรคูณจากแม่ 2-12
8. จงเปรียบเทียบความเหมือนและแตกต่างของโปรแกรมภาษาซีและภาษาจาวา
9. จงอธิบายและยกตัวอย่างคำสวณของภาษาจาวามาอย่างน้อย 10 คำ
10. จงเปรียบเทียบข้อดีข้อเสียของการใช้งานภาษาจาวาที่เป็นโอเพ่นซอร์สกับโปรแกรมไมโครซอฟท์วิซวลเบสิกดอทเน็ตที่ต้องซื้อ

บทที่ 2

คำสั่งพื้นฐานของภาษาจาวา

(Basic Java Commands)

วัตถุประสงค์

เพื่อให้ผู้เรียน

1. มีความรู้และสามารถใช้งานคำสั่งแสดงผลมาตรฐานของโปรแกรมภาษาจาวาได้
2. มีความรู้ และสามารถใช้งานคำสั่งนำเข้ามาตรฐานของโปรแกรมภาษาจาวาได้
3. มีความรู้และสามารถใช้งานตัวดำเนินการและคำสั่งดำเนินการของโปรแกรมภาษาจาวาได้

การเขียนโปรแกรมคอมพิวเตอร์ เริ่มจากการเขียนรหัสต้นฉบับให้ถูกต้องตามรูปแบบที่กำหนดของไวยากรณ์ของโปรแกรมภาษา หลังจากนั้นคอมไพเลอร์โปรแกรมเพื่อแปลคำสั่งจากรหัสต้นฉบับให้อยู่ในรูปแบบของคอมพิวเตอร์เข้าใจและหากไม่มีข้อผิดพลาดโปรแกรมจะสามารถรันได้ทันที นอกจากนั้นผู้เขียนโปรแกรมต้องมีความรู้ด้านการวิเคราะห์และออกแบบโปรแกรมตามแนวทางของอัลกอริทึม (Algorithm) เพื่อออกแบบและเขียนโปรแกรมให้ได้ตามที่ผู้ใช้งานต้องการ การฝึกเขียนโปรแกรมที่ง่ายที่สุด คือ การเริ่มสั่งให้คอมพิวเตอร์แสดงผลทางจอภาพให้ได้ผลดังที่ผู้เขียนต้องการด้วยคำสั่งแสดงผลพื้นฐาน ส่วนมากจะเริ่มด้วยการสั่งให้คอมพิวเตอร์แสดงผล ลำดับต่อมาคือการนำเข้าข้อมูลโดยใช้คำสั่งรับข้อมูล แล้วนำข้อมูลดังกล่าวไปประมวลผลหรือแสดงผลต่อไป หลังจากการนำเข้าข้อมูล จำเป็นต้องคำนวณ ในเชิงเทคนิคการเขียนแล้วเรียกกลุ่มรายการคำนวณดังกล่าวว่านิพจน์ (Expressions) ซึ่งอาจเป็นค่าแบบไม่ซับซ้อนหรือเป็นเซตของตัวดำเนินการ โดยที่ตัวดำเนินการ (Operator) เป็นสัญลักษณ์ที่ใช้สำหรับสั่งให้คอมพิวเตอร์นำเอาข้อมูลที่ได้ออกไปประมวลผล โดยต้องระบุสัญลักษณ์ของการคำนวณในลักษณะแตกต่างกันตามวัตถุประสงค์ของการคำนวณ

บทนี้แนะนำคำสั่งที่ใช้สำหรับการแสดงผลเบื้องต้น คำสั่งสำหรับการนำเข้าข้อมูล ตัวแปรสำหรับรับข้อมูล และตัวดำเนินการต่างๆ กับข้อมูลที่นำเข้าและนำไปแสดงผล เพื่อให้ผู้เรียนได้มีความรู้เรื่องรูปแบบการเขียนโปรแกรมเบื้องต้นจาก การแสดงผล การนำเข้า และการดำเนินการสำหรับคำนวณต่างๆ

การแสดงผลมาตรฐาน

คำสั่งแสดงผลมาตรฐานที่ใช้สำหรับภาษาจาวานั้นจะใช้งานคลาส System ด้วยการเรียกใช้อ็อบเจกต์ out และเรียกใช้งานผ่านเมธอด print และ println โดยใช้จุด (.) เป็นตัวแบ่งการเรียกใช้งาน เช่น System.out.println เป็นต้น ต่อไปนี้เป็น การแสดงผลการใช้งานเมธอดทั้งสองโดยละเอียดและขณะเดียวกันจะนำเสนอตัวอย่างการใช้งานของเมธอดดังกล่าวในรูปแบบแตกต่างกัน

คำสั่ง System.out.print

คำสั่ง System.out.print นี้สามารถแสดงผลได้ทั้งการผ่านค่าพารามิเตอร์ที่เป็นอ็อบเจกต์ ข้อความ