

โครงสร้างข้อมูล ด้วยภาษาจาวา

Data Structures By Java ADT's

ArrayList, Stack, Queue

LinkedList, Tree, Hashtable

Dictionary, String, Collection&Collections

โดย ศูนย์หนังสือราคานักเรียน

295 บาท



โครงสร้างข้อมูลด้วยภาษาจาวา

Data Structures by Java ADT's

Document Date: ธันวาคม 2564, Number #1, Volume: 01

Author: ศูนย์หนังสือราคานักเรียน

ADDRESS INFORMATION

สามารถติดต่อ ผู้เขียนได้โดยผ่าน อีเมลเท่านั้น

Email: sk_aran01@yahoo.com

คำนำ

วัตถุประสงค์หลักของหนังสือเล่มนี้ เพื่อใช้โครงสร้างข้อมูลที่มีอยู่ในภาษาจาวา ที่ Array Stack Queue Linked Lists Tree Hashing Dictionary Collection Sorting Searching และ String ตามแนวทางของโครงสร้างข้อมูล เน้นการใช้งานโดยอาศัยคลาสชนิดข้อมูลนามธรรม Java Abstract Data Types (Java ADT's) ที่มีอยู่แล้ว โดยเน้นการสร้าง การดำเนินการ และการเข้าถึงข้อมูลต่างๆ ตามแนวทางตัวดำเนินการของโครงสร้างข้อมูล ซึ่งต้องการสร้างโครงสร้างและการสร้างโค้ดการเขียนโปรแกรมเพื่อทำการดำเนินการ เพียงแค่ประกาศอ็อบเจกต์และเรียกใช้เมธอดต่างๆ ในนั้นก็สามารถใช้งานได้ทันที ซึ่งจะลดเรื่องของโค้ด รับประกันการถูกต้องของอัลกอริทึมอีกด้วย

ผู้เขียนหวังว่าผู้ศึกษาหนังสือเล่มนี้จะได้ประโยชน์ตามวัตถุประสงค์ที่ตั้งเอาไว้ และมีความรู้ความเข้าใจสำหรับนำเนื้อหาความรู้ไปสู่การพัฒนาตนเองเป็นนักวิทยาการคอมพิวเตอร์หรือนักเทคโนโลยีคอมพิวเตอร์ที่สมบูรณ์ต่อไป ขอขอบคุณ "Free download by <https://usedtotech.com>" สำหรับแม่แบบ (Template) สำหรับทำเล่มหนังสือนี้

ศูนย์หนังสือราคานักเรียน

23 ธันวาคม 2564

สารบัญ

บทที่ 1: แนะนำโครงสร้างข้อมูล	7
➤ ชนิดข้อมูลนามธรรม	
➤ ข้อมูลนามธรรมในภาษาจาวา	
➤ การดำเนินการชนิดข้อมูลนามธรรมในภาษาจาวา	
➤ เนื้อหาที่จะนำเสนอในเล่ม	
➤ สรุป	
➤ แบบฝึกหัดท้ายบท	
บทที่ 2: อาร์เรย์ลิสต์	15
➤ อาร์เรย์ลิสต์ (java.util.ArrayList)	
➤ เวกเตอร์ (java.util.Vector)	
➤ สรุป	
➤ แบบฝึกหัดท้ายบท	
บทที่ 3: สแต็ก	31
➤ ความรู้เบื้องต้นเกี่ยวกับสแต็ก	
➤ การใช้งานสแต็ก (java.util.Stack)	
➤ การประยุกต์ใช้งาน	
➤ สรุป	
➤ แบบฝึกหัดท้ายบท	
บทที่ 4: คิว	44
➤ ความรู้เบื้องต้นเกี่ยวกับคิว	
➤ การใช้งานคิวทั่วไป	
➤ คิวที่มีความสำคัญ	
➤ คิวแบบสองทาง	
➤ สรุป	
➤ แบบฝึกหัดท้ายบท	

บทที่ 5: ลิงค์ลิสต์

61

- ความรู้เบื้องต้นเกี่ยวกับลิงค์ลิสต์
- การใช้งานลิงค์ลิสต์
- ลิสต์
- สรุปรูป
- แบบฝึกหัดท้ายบท

บทที่ 6: ทรี

73

- ความรู้เบื้องต้นเกี่ยวกับทรี
- ทรีเซต
- ทรีแมป
- สรุปรูป
- แบบฝึกหัดท้ายบท

บทที่ 7: แอชเทเบิล

85

- ความรู้เบื้องต้นเกี่ยวกับแอชเทเบิล
- แอชเทเบิล
- แอชแมป
- แอชเซต
- สรุปรูป
- แบบฝึกหัดท้ายบท

บทที่ 8: ดิกชันนารี

104

- การใช้งานดิกชันนารี
- เทคนิคการใช้งานดิกชันนารี
- สรุปรูป
- แบบฝึกหัดท้ายบท

บทที่ 9: สตริง

115

- ความรู้เบื้องต้นเกี่ยวกับสตริง
- การใช้งานสตริง
- สรุปรูป
- แบบฝึกหัดท้ายบท

บทที่ 10: คอลเลกชัน

126

- ความรู้เบื้องต้นเกี่ยวกับคอลเลกชัน
- การใช้งานอ็อบเจกต์คอลเลกชัน
- การใช้งานคลาสสแตตติกคอลเลกชัน
- การประยุกต์ในการจัดเรียงและการค้นหา
- สรุป
- แบบฝึกหัดท้ายบท

บรรณานุกรม

147

บทที่ 1 : แนะนำโครงสร้างข้อมูล

(Introduction to Java abstract Data Type's)

เนื้อหา

ชนิดข้อมูลนามธรรม

ข้อมูลนามธรรมในภาษาจาวา

การดำเนินการชนิดข้อมูลนามธรรมในภาษาจาวา

เนื้อหาที่จะนำเสนอในเล่ม

สรุป

แบบฝึกหัดท้ายบท

ชนิดข้อมูลนามธรรม

ชนิดข้อมูลนามธรรม (Abstract Data Type: ADT) คือ การนำแนวคิดของการกำหนดสาระสำคัญของข้อมูล (Data Abstraction) เพื่อเสนอโครงสร้างข้อมูลในรูปแบบเชิงตรรกะ (Logical Form) ที่ไม่ยึดติดกับฮาร์ดแวร์หรือซอฟต์แวร์ใด จะนำไปใช้เป็นพื้นฐานในการพัฒนาโปรแกรมแนวคิดการสร้างโครงสร้างข้อมูลชนิดข้อมูลนามธรรม โดยทั่วไปมักจะมีตัวดำเนินการหรืออัลกอริทึมให้เรียกใช้ได้อัตโนมัติ เช่น Create, Insert,

Delete, Traversal, Search, Sort เป็นต้น ชนิดข้อมูลนามธรรมช่วยซ่อนรายละเอียดภายในของการทำงานในโครงสร้างข้อมูลประเภทต่างๆ โดยเฉพาะข้อมูลประเภทซับซ้อน (Compound Data Structure) เช่น รายการแถวลำดับ (ArrayLists) กองซ้อน (Stack) แถวคอย (Queue) รายการโยง (LinkedLists) เป็นต้น

เมื่อชนิดข้อมูลนามธรรมถูกนำมาใช้งาน ไม่จำเป็นต้องมีรายละเอียดของอัลกอริทึมมาเขียนแทรกเข้าไปในโปรแกรมคอมพิวเตอร์ แต่ทั้งนี้ หากโปรแกรมภาษาใดจะนำไปใช้งานจะต้องสร้างให้ได้ตามที่รูปแบบของชนิดข้อมูลนามธรรมนั้นๆ ด้วย

1) การประกาศข้อมูล (Declaration of Data)

2) ประกาศตัวดำเนินการ (Declaration of Operations) และ

3) การหุ้มของการดำเนินการข้อมูล (Encapsulation of Data Operations) ซึ่งซ่อนรายละเอียดของอัลกอริทึมเอาไว้ ผู้ใช้งานหรือโปรแกรมเมอร์ เพียงแต่ประกาศตัวแปรข้อมูลชนิดนามธรรมชนิดนั้นๆ ขึ้นมา และเรียกใช้งานตัวดำเนินการดังกล่าว ที่ระบุไว้เสมือนเรียกใช้ฟังก์ชันหรือเมธอดอัตโนมัติได้ ทำให้ได้ผลลัพธ์ที่ต้องการได้อย่างรวดเร็วและสะดวกมากยิ่งขึ้น ในปัจจุบันพบมากในการเขียนโปรแกรมเชิงวัตถุ เช่น ภาษาจาวา ภาษาในตระกูลของวิซวลดอตเน็ต ไพธอน เป็นต้น

สำหรับในหนังสือเล่มนี้ แสดงอัลกอริทึมต่างๆ ด้วยโปรแกรมภาษาจาวา ดังนั้น แต่ละบทที่เกี่ยวข้องกับโครงสร้างข้อมูลประเภทต่างๆ ผู้เขียนได้นำเสนอหัวข้อของชนิดข้อมูลนามธรรมชนิดนั้นๆ ที่มีอยู่ในโปรแกรมภาษาจาวาอีกด้วย

ชนิดข้อมูลนามธรรมในจาวา

ชนิดข้อมูลนามธรรมในจาวา (Java Abstract Data Types: Java ADT's) คือ ชุดของคลาสที่อยู่ในแพ็คเกจ `java.util` ที่ให้ผู้ใช้งาน (โปรแกรมเมอร์) สามารถ import มาใช้ในโปรแกรมของตน แล้วสร้างอ็อบเจกต์ (Create Object) หลังจากนั้น อ็อบเจกต์ต่างจะมีตัวดำเนินการ (Operators) ให้สามารถใช้งาน และมีการหุ้ม (Encapsulation) ของอัลกอริทึมของตัวดำเนินการต่างๆ ให้เรียบร้อย มีการจัดการการเปลี่ยนแปลงต่างๆ เมื่อมีการเพิ่ม ลบ แก้ไข และดำเนินการใดๆ กับโครงสร้างข้อมูลที่สร้างขึ้นมาใช้งาน โดยปกติจะมีโครงสร้างข้อมูลมาตรฐานที่ใช้ในการศึกษาในหลักการวิทยาการคอมพิวเตอร์ เช่น ArrayList List, Stack, Queue, Set, Map Collection เป็นต้น

การดำเนินการชนิดข้อมูลนามธรรมในจาวา

หลักการใช้งานและดำเนินการกับข้อมูลชนิดนามธรรมของจาวา จะมีการดำเนินการอยู่ 4 ลักษณะดังนี้

1. ตัวสร้าง (Creators) คือการดำเนินการเพื่อสร้างอ็อบเจกต์เมื่อมีการ import คลาสหรืออินเทอร์เฟสเข้ามา ซึ่งจะมีรูปแบบการสร้างอ็อบเจกต์ที่หลากหลายตามแนวทางการเขียนโปรแกรมแบบเชิงวัตถุด้วยภาษาจาวา
2. ตัวดำเนินการ (Producers) ตัวดำเนินการจะมีมาให้อัตโนมัติโดยแสดงผ่านเมธอดการใช้งานต่างๆ ของอ็อบเจกต์ที่ประกาศใช้งานในโปรแกรม แต่ละชนิดของโครงสร้างข้อมูลที่จะประกาศใช้งานจะมีตัวดำเนินการที่แตกต่างกันไป ตามแนวทางของโครงสร้างข้อมูลพื้นฐาน
3. ตัวตรวจจับคุณลักษณะของโครงสร้าง (Observers) เป็นคุณลักษณะที่โครงสร้างข้อมูลชนิดที่ถูกสร้างเป็นอ็อบเจกต์แล้วนั้น จะมีคุณลักษณะประจำตัวให้อัตโนมัติ สามารถเข้าถึงผ่านเมธอดต่างๆ หรือแอตทริบิวต์ประจำของอ็อบเจกต์ที่ถูกสร้างขึ้น
4. การเปลี่ยนแปลงได้เอง (Mutators) เมื่อมีการเปลี่ยนแปลงในโครงสร้างขึ้น ไม่ว่าจะเป็นการเพิ่ม ลบ หรือ แก้ไขข้อมูลในโครงสร้างที่บรรจุข้อมูลที่บรรจุไว้ การจัดการโครงสร้างดังกล่าวจะเกิดขึ้นเองอัตโนมัติโดยผู้ใช้งานหรือโปรแกรมเมอร์ไม่ต้องจัดการด้วยตนเอง

แสดงตัวอย่างชนิดข้อมูลนามธรรมของภาษาจาวา จากแพ็คเกจ

java.util.ArrayList ดังนี้

ตัวอย่างที่ 1.1 กำหนดตัวแปร A เป็นโครงสร้างข้อมูลชนิดข้อมูลนามธรรมประเภท ArrayList จากแพ็คเกจ java.util.ArrayList ของโปรแกรมภาษาจาวา จงแสดงตัวอย่างตัวดำเนินการที่อยู่ในตัวแปร A ที่ผู้ใช้งานสามารถเรียกใช้งานได้มาพอสังเขป

แนวคิด : เบื้องต้นต้อง import java.util.ArrayList แล้ว ประกาศตัวแปร A ในโปรแกรม เช่น ArrayList A = new ArrayList(); จะทำให้ได้ตัวแปร A ดังที่โจทย์ต้องการ หรือ อาจประกาศโดยตรงเช่น java.util.ArrayList A = new java.util.ArrayList(); ก็จะได้ตัวแปรดังโจทย์เช่นกัน จากนั้น ในโปรแกรม พิมพ์ A. ซึ่งจะมีตัวดำเนินการต่างๆ ของ ข้อมูลชนิดนามธรรมประเภท ArrayList นี้มาให้ใช้งาน แสดงดังรูป

*add(Object e)	boolean	*lastIndexOf(Object o)	int
*add(int i, Object e)	void	*listIterator()	ListIterator
*addAll(Collection clctn)	boolean	*listIterator(int i)	ListIterator
*addAll(int i, Collection clctn)	boolean	*notify()	void
*clear()	void	*notifyAll()	void
*clone()	Object	*parallelStream()	Stream
*contains(Object o)	boolean	*remove(Object o)	boolean
*containsAll(Collection clctn)	boolean	*remove(int i)	Object
*ensureCapacity(int i)	void	*removeAll(Collection clctn)	boolean
*equals(Object o)	boolean	*removeIf(Predicate prdct)	boolean
*forEach(Consumer cnsmr)	void	*replaceAll(UnaryOperator uo)	void
*get(int i)	Object	*retainAll(Collection clctn)	boolean
*getClass()	Class<?>	*set(int i, Object e)	Object
*hashCode()	int	*size()	int
*indexOf(Object o)	int	*sort(Comparator cmptr)	void
*isEmpty()	boolean	*splitter()	Splitter
*iterator()	Iterator	*stream()	Stream

รูปที่ 1.1 แสดงตัวอย่างตัวดำเนินการใน java ArrayList ADT

เนื้อหาที่จะนำเสนอในเล่ม

เนื้อหาที่เรียกใช้จะมาจาก แพ็กเกจ `java.util` ซึ่งมีโครงสร้างข้อมูลดังนี้

- `ArrayList` และ `Vector`
- `Stack`
- `Queue`
- `LinkedList`, `List`
- `Tree`, `TreeSet` `TreeMap`,
- `Hashtable`, `HashSet`, `HashMap`,
- `Dictionary`,
- `Collection`, `Collections`

สรุป

โครงสร้างข้อมูลชนิดข้อมูลนามธรรมเป็นการนำเสนอโครงสร้างข้อมูลในรูปแบบเชิงตรรกะสำหรับโปรแกรมชั้นสูงเพื่อให้ผู้เขียนโปรแกรมได้ใช้งานโครงสร้างข้อมูลต่างๆ ได้สะดวกรวดเร็ว ไม่จำเป็นต้องคำนึงถึงอัลกอริทึมที่อยู่ภายใน สำหรับโครงสร้างข้อมูลชนิดข้อมูลนามธรรมของภาษาจาวา จะสร้างอ็อบเจกต์ของโครงสร้างข้อมูลที่ต้องการใช้งานจากนั้นดำเนินการผ่านเมธอดต่างๆ ที่โครงสร้างข้อมูลชนิดนั้นๆ ที่โปรแกรมภาษาจาวาจัดเตรียมไว้ให้ใช้งาน โดยเรียกใช้จากแพ็คเกจ `java.util` ตามด้วยชื่อโครงสร้างข้อมูลที่ต้องการใช้งาน