

# ใบงานการทดลองที่ 3

## เรื่อง อาร์เรย์ สตริง และฟังก์ชัน ในภาษาจาวา

### 1. จุดประสงค์ทั่วไป

- 1.1. รู้และเข้าใจการโปรแกรมเชิงวัตถุร่วมกับอาร์เรย์และสตริง
- 1.2. รู้และเข้าใจการโปรแกรมเชิงวัตถุร่วมกับฟังก์ชัน

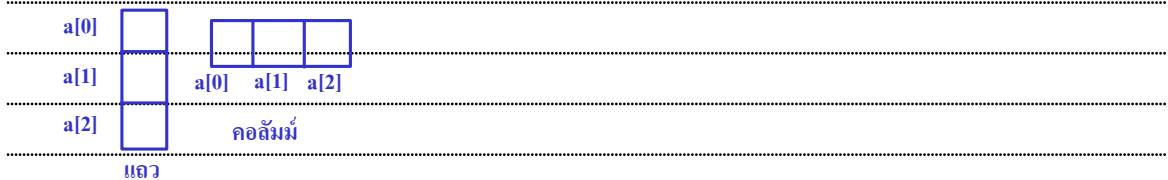
### 2. เครื่องมือและอุปกรณ์

เครื่องคอมพิวเตอร์ 1 เครื่อง ที่ติดตั้งโปรแกรม Eclipse

### 3. ทฤษฎีการทดลอง

- 3.1. โครงสร้างข้อมูลแบบ “อาร์เรย์” มีลักษณะเป็นอย่างไร ? มีองค์ประกอบอะไรบ้าง ? อธิบายพร้อมยกตัวอย่างประกอบ

ข้อมูลในลักษณะของการจองพื้นที่หน่วยความจำติดกัน โดยจำนวนข้อมูลลงที่ และแต่ละข้อมูลต้องเป็นชนิดเดียวกัน



- 3.2. การเข้าถึงแต่ละ Element ของอาร์เรย์สามารถทำได้อย่างไร ? อธิบายพร้อมยกตัวอย่างประกอบ  
โดยอ้างอิงจากหมายเลขของ index

```
String[] cars = {"volvo", "BMW", "Ford", "Mazda"};  
System.out.println(cars[0]);
```

//ผลลัพธ์ที่ได้ volvo

- 3.3. คำสั่ง length เกี่ยวข้องกับอาร์เรย์อย่างไร ? อธิบายพร้อมยกตัวอย่างประกอบ

ใช้นับจำนวนของ element

```
String[] cars = {"volvo", "BMW", "Ford", "Mazda"};  
System.out.println(cars.length);
```

//ผลลัพธ์ที่ได้ 4

- 3.4. จงยกตัวอย่างประกอบในการวนรอบเพื่อแสดงค่าภายในตัวแปรอาร์เรย์ตั้งแต่ค่าแรกจนถึงค่าสุดท้าย

```
String[] cars = {"volvo", "BMW", "Ford", "Mazda"};  
for (String i : cars){  
    System.out.println(i);  
}
```

- 3.5. จงยกตัวอย่างการใช้งานคำสั่ง for each เพื่อแสดงค่าภายในตัวแปรอาร์เรย์

```
String[] cars = {"volvo","BMW","Ford","Mazda"};
for (String i : cars){
System.out.println(i);
}
```

3.6. เหตุใดจึงต้องมีคำสั่ง import java.util.Arrays ; ในส่วนต้นของไฟล์ ?

3.7. คำสั่ง Arrays.copyOf( \_\_\_\_\_ , \_\_\_\_\_ ) ; มีหน้าที่ทำอะไร ?

ใช้ลดหรือเพิ่ม array

โดยในวงเล็บจะประกอบด้วย ( array ต้นฉบับ , array ใหม่ )

3.8. จงยกตัวอย่างการประกาศ String และกำหนดค่าคำว่า “Hello World” ในภาษาจาวา

```
String txt = "Hello World";
System.out.println( txt );
```

3.9. จงอธิบายและยกตัวอย่างประกอบการใช้งานคำสั่ง toUpperCase() ในภาษาจาวา

แปลง String เป็นตัวพิมพ์ใหญ่

```
String txt = "hello world"; //Output HELLO WORLD
System.out.println( txt . toUpperCase());
```

3.10. จงอธิบายและยกตัวอย่างประกอบการใช้งานคำสั่ง toLowerCase() ในภาษาจาวา

แปลง String เป็นตัวพิมพ์เล็ก

```
String txt = "HELLO WORLD"; //Output hello world
System.out.println( txt . toLowerCase());
```

3.11. จงอธิบายและยกตัวอย่างประกอบการใช้งานคำสั่ง indexOf() ในภาษาจาวา

คือ method จะสั่ง indexOf() ของการ	int indexOf(String str)	_____	ค่า string แทน string ที่ต้องการค้นหา
เกิดครั้งแรกของตัวอักษรที่ระบุใน String	int indexOf(String str, int fromIndex)	_____	แสดงตำแหน่งที่จะเริ่มต้นการค้นหา
	int indexOf(int char)	_____	แสดงอักษรตัวเดียวที่จะเริ่มต้นการค้นหา

3.12. จงอธิบายความแตกต่างระหว่างการเชื่อม String แบบปกติและแบบใช้คำสั่ง concat( )

คือ method สำหรับเชื่อมต่อ

string concat() ตามด้วย string อีกตัว

```
String firstName = "John ";
String lastName = "Doe"; //Output John Coc
System.out.println(firstName.concat(lastName));
```

3.13. หากต้องการแสดงสัญลักษณ์พิเศษภายในตัวแปร String ควรทำอย่างไร ?

```
String txt = " What's UP";
System.out.println( txt );
```

3.14. จงอธิบายและยกตัวอย่างประกอบการสร้างฟังก์ชันในภาษาจาวา

```
public static void main (String[] args){
```

```
}
```

3.15. อธิบายข้อแตกต่างระหว่าง Pass by value และ Pass by reference

3.16. ความแตกต่างระหว่างการประกาศฟังก์ชันแบบ void กับแบบ int, double, float, string คืออะไร ?

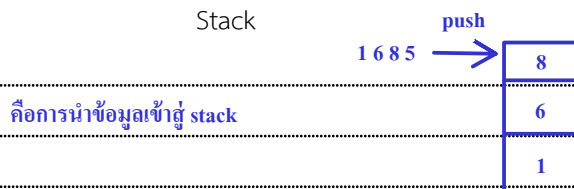
**void จะไม่ส่งค่ากลับคืน**

3.17. โครงสร้างข้อมูลแบบ Stack แตกต่างกับ Array อย่างไร ?

**stack จะเรียงแบบ เข้าก่อนออกหลังซึ่งไม่สามารถแทรกกลางได้ ส่วน array จะเรียงไปเรื่อยๆ array สามารถแทรกตรงกลางได้**

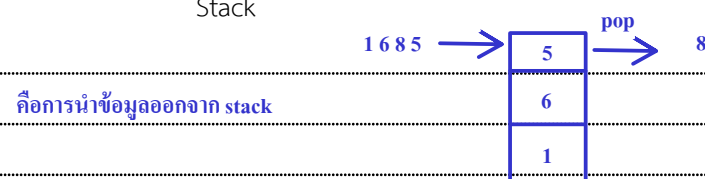
3.18. อธิบายพร้อมยกตัวอย่างประกอบกระบวนการทำงานของคำสั่ง Push ในโครงสร้างข้อมูลแบบ

Stack



3.19. อธิบายพร้อมยกตัวอย่างประกอบกระบวนการทำงานของคำสั่ง Pop ในโครงสร้างข้อมูลแบบ

Stack



3.20. อธิบายพร้อมยกตัวอย่างประกอบกระบวนการทำงานของคำสั่ง isEmpty ในโครงสร้างข้อมูลแบบ

Stack

การตรวจสอบข้อมูล stack ว่ามีที่ว่างหรือไม่

```
Stack.add 1;  
Stack.add 5;  
Stack.add 1;  
System.out.println(" st : "+Stack );  
System.out.println(" Is Empty "+Stack.isEmpty() );
```

3.21. อธิบายพร้อมยกตัวอย่างประกอบความหมายของคำว่า Stack overflow

Stack เต็ม

#### 4. ลำดับขั้นการปฏิบัติการ

4.1. จงแก้โจทย์ปัญหาดังต่อไปนี้

4.1.1. จงเขียนโปรแกรมเพื่อสุ่มค่าเข้าไปในอาร์เรย์ 1 มิติ ตามจำนวนค่าที่รับจากผู้ใช้งาน โดยค่าที่ถูกสุ่มจะต้องเป็นตัวเลขจำนวนเต็มที่อยู่ระหว่าง 0 ถึง 99 เท่านั้น

Test case 1	Test case 2
Please enter your random value : <b>8</b> ----- Array[0] = 94 Array[1] = 32 Array[2] = 46 Array[3] = 18 Array[4] = 27 Array[5] = 5 Array[6] = 31 Array[7] = 17	Please enter your random value : <b>12</b> ----- Array[0] = 56 Array[1] = 27 Array[2] = 13 Array[3] = 15 Array[4] = 65 Array[5] = 29 Array[6] = 11 Array[7] = 92 Array[8] = 95 Array[9] = 47 Array[10] = 58 Array[11] = 62

4.1.2. ผังงานแสดงกระบวนการทำงานและโค้ดโปรแกรม(ที่ตรงตามผังงาน)

ผังงาน	โค้ดโปรแกรม
	<pre>1 package lab3; 2 import java.util.Random; 3 import java.util.Scanner; 4 public class lab3 { 5 6     private static Scanner sc; 7     public static void main(String[] args) { 8         int n,i; 9         sc = new Scanner(System.in); 10        System.out.print(" Please Enter your random value : "); 11        n = sc.nextInt(); 12        int [] arr = new int[n]; //สร้าง array มาจาก n 13 14        for ( i = 0; i &lt; arr.length; i++) { 15            arr[i] = new Random().nextInt(99); // เก็บเลขสุ่มมา ลงใน array i ของ 16            for(int j = 0; j &lt;= arr.length ; j++) //วน จำนวนของ array เป็น n 17            { 18                if (i != n) { 19                    System.out.println("Array"+"["+i+ " " +arr[i]); 20                    break; 21                } 22            } 23        } 24    } 25 } 26</pre>

--	--

--	--

#### 4.2. จงแก้ไขข้อปัญหาดังต่อไปนี้

##### 4.2.1. จงเขียนฟังก์ชันการจัดการโครงสร้างข้อมูลแบบ Stack พร้อมจำลองการทำงานโดยการเรียกใช้คำสั่งพื้นฐานดังต่อไปนี้

คำสั่ง Push( String Value ) ; เพื่อนำข้อมูลเข้าไปเก็บไว้ใน Stack

คำสั่ง Pop( ) ; เพื่อนำข้อมูลบนสุดออกจาก Stack

คำสั่ง isEmpty( ) ; เพื่อตรวจสอบข้อมูลใน Stack ว่ามีอยู่หรือไม่

คำสั่ง Top( ) ; เพื่อตรวจสอบข้อมูลที่อยู่บนสุด

คำสั่ง CheckStack( ) ; เพื่อตรวจสอบค่าภายใน Stack ทั้งหมด

คำสั่ง SetStackSize( int size ) ; เพื่อกำหนดขนาดเริ่มต้นของ Stack

Test case
SetStackSize( 3 ) isEmpty ----  Yes Top ----  NULL Push : Hello CheckStack ----  STACK : Hello Push : Hi CheckStack ----  STACK : Hi, Hello Push : Test CheckStack ----  STACK : Test, Hi, Hello Top ----  Top = Test Pop CheckStack

```
----| STACK : Hi, Hello  
isEmpty  
----| No  
Push : OK  
CheckStack  
----| STACK : OK, Hi, Hello  
Push : RMUTL  
----| Stack Overflow  
CheckStack  
----| STACK : OK, Hi, Hello
```



#### 4.2.2. ผังงานแสดงกระบวนการทำงานและโค้ดโปรแกรม(ที่ตรงตามผังงาน)

ผังงาน	โค้ดโปรแกรม
	<pre> 1 package lab3_2; 2 import java.util.*; 3 4 5 public class lab3_2 { 6     private String[] stack; 7     private int size; 8     private int top; 9 10    public lab3_2(int size) { 11        this.size = size; 12        stack = new String[size]; 13        top = -1; 14    } 15 16    // Method to add a value to the stack 17    public void push(String value) { 18        if (top == size - 1) { 19            System.out.println("\n----  Stack: Overflow"); 20        } else { 21            top++; 22            stack[top] = value; 23            System.out.println("\nPush: " + stack[top]); 24        } 25    } 26 27    // Method to remove the top element from the stack 28    public String pop() { 29        if (top == -1) { 30            System.out.println("----  NULL"); 31            return null; 32        } else { 33            String value = stack[top]; 34            top--; 35            System.out.println("Pop"); 36            return value; 37        } 38    } 39 40    // Method to check if the stack is empty 41    public boolean isEmpty() { 42        System.out.println("\nisEmpty"); 43        if (top == -1) { 44            System.out.println("----  Yes"); 45            return true; 46        } else { 47            System.out.println("----  No"); 48            return false; 49        } 50    } 51 52    // Method to retrieve the top element of the stack 53    public String top() { 54        System.out.println("\nTop"); 55        if (top == -1) { 56            System.out.println("----  NULL"); 57            return null; 58        } else { 59            System.out.println("----  Top = " + stack[top]); 60            return stack[top]; 61        } 62    } 63 64    // Method to check the current stack size 65    public void checkStack() { 66        System.out.println("\nCheckStack"); 67        if (top == -1) { 68            System.out.println("----  Stack : NULL"); 69        } else { 70            for (int i = top; i &gt;= 0; i--) { 71                System.out.printf(stack[i]); 72                if (i &gt; 0) { 73                    System.out.print(", "); 74                } 75            } 76        } 77    } </pre>

```

77 // Method to set the maximum size of the stack
78 public void setStackSize(int size) {
79     System.out.println("SetStackSize("+size+"");
80     this.size = size;
81     stack = new String[size];
82     top = -1;
83 }

```

```

114 public static void main(String[] args) {
115     Scanner sc = new Scanner(System.in);
116     lab3_2 stack = new lab3_2(10); // Initialize stack with maximum size of 10
117
118     while (true) {
119         System.out.println("Enter a command (push, pop, isEmpty, top, checkStack, setStackSize, or exit): ");
120         String input = sc.nextLine();
121
122         switch (input) {
123             case "push":
124                 System.out.println("Enter a value to push onto the stack: ");
125                 String value = sc.nextLine();
126                 stack.push(value);
127                 break;
128             case "pop":
129                 stack.pop();
130                 break;
131             case "isEmpty":
132                 stack.isEmpty();
133                 break;
134             case "top":
135                 stack.top();
136                 break;
137             case "checkStack":
138                 stack.checkStack();
139                 break;
140             case "setStackSize":
141                 System.out.println("Enter a new stack size:");
142                 String sizeInput = sc.nextLine();
143                 int size = Integer.parseInt(sizeInput);
144                 stack.setStackSize(size);
145                 break;
146             case "exit":
147                 sc.close();
148                 return;
149             default:
150                 System.out.println("Invalid command. Please try again.");
151                 break;
152         }
153     }
154 }

```

```

121 break;
122 }
123 }
124 }
125 }

```

--	--

## 5. สรุปผลการปฏิบัติการ

จากการทดลองพบว่าในโปรแกรมสุ่มค่าต่างเข้าไปใน Array จะมีการใช้ Package Random และองค์ประกอบหลักของ Stack คือ Push , peek โดยจากการทดลองจะเรียกใช้งาน package Stack เพื่อดำเนินการต่างๆ

## 6. คำถามท้ายการทดลอง

6.1. ข้อควรระวังในการใช้งาน Array ในภาษาจาวาคืออะไร ?

หากต้องการจัดเก็บองค์ประกอบจำนวนมาใน Array ให้ใช้ ArrayList เพราะ เป็น Array ที่ปรับขนาดได้

6.2. ข้อควรระวังในการใช้งาน String ในภาษาจาวาคืออะไร ?

หากต้องการแก้ไขสตริง ควรใช้ StringBuilder หรือ StringBuffer แทน, ใช้ equals() เพื่อเปรียบเทียบเนื้อหาของวัตถุสตริงสองรายการ

6.3. ฟังก์ชันในภาษาจาวาไม่สามารถใช้งานแบบ Pass by reference ในภาษาซีได้ คุณมีแนวทางการแก้ไขปัญหานี้ได้อย่างไร ?

การ Pass by reference จะต้องใช้คีย์เวิร์ด ref ในการส่งตัวแปรไปยัง function โดยตัวแปรที่ถูกส่งผ่านจะส่งเป็นอาร์กิวเมนต์โดยจะต้องกำหนด ref

6.4. โครงสร้างข้อมูลแบบ Stack แตกต่างกับโครงสร้างข้อมูลแบบ Array อย่างไร ?

สแต็คมีขนาดที่คงที่โดยต้องระบุจำนวนสูงสุดที่จะเก็บไว้ กลับกัน อาร์เรย์สามารถปรับขนาดได้แบบไดนามิก ในการแก้ไของค์ประกอบในสแต็คสามารถใช้การ push, pop ได้แต่ในอาร์เรย์จะต้องใช้ indices ในการแก้ไข

