



# Programming Fundamentals II

Conditional and Collection

# Let's recap the previous session

- Boolean expressions
- Short-circuit evaluation
- Floating-point comparison
- Conditional expressions
- Block statements
- Block scope
- `if/if-else`
- `if-else-if` ladder
- `switch`
- Multiple labels and fall-through
- `while`
- `do-while`
- `break` and `continue`
- Pre- and post-increment and decrement
- Compound assignment

The two  
forms of if

---

```
if (condition)  
    true-branch
```

---

```
if (condition)  
    true-branch  
else  
    false-branch
```

The switch statement

```
switch (expression) {  
    case constant1:  
        branch1  
    case constant2:  
        branch2  
  
    ...  
    default:  
        branchn  
}
```

This part is optional



# The while loop

```
while (condition)  
    body
```

- Check, then act



# The do ... while loop

do

*body*

while (*condition*);

- Act, then check

Let's continue our  
journey...





# The for loop

```
for (control_variable = value;  
    boolean_expression;  
    increment or decrement)  
{  
    body;  
}
```



Python

```
for i in range(4):  
    for j in range(i+1):  
        print "*",  
    print
```

Java

```
for (int i = 0; i < 4; i++) {  
    for (int j = 0; j < i + 1; j++)  
        System.out.print("*");  
    System.out.println();  
}
```

# Collections

1. Scanner/Format Class
2. Arrays
3. Strings
4. ArrayLists
5. Looping over a collection
6. enum
7. HashMap

# Input/Scanner Class

มี 2 Class ให้เลือกใช้

1. InputStreamReader
2. Scanner

## InputStreamReader

- import แพคเกจ java.io
- รับได้เฉพาะ String เท่านั้น
- นิยมใช้กับ system input อื่นที่ไม่ใช่

System.in

## Scanner

- import แพคเกจ java.util.Scanner
- รับข้อความ หรือตัวเลขได้
- สร้าง 1 Object เพื่อใช้งาน

# Streaming input from standard input

```
15 char[] bb = new char[100];
16 try (InputStreamReader ipsr = new InputStreamReader(System.in, "UTF-8")) {
17     int aa;
18     System.out.print("Enter string: ");
19     //ipsr.read(aa); //Causing compile error
20     ipsr.read(bb);
21     System.out.printf("The result of InputStreamReader class B = %s%n", new String(bb));
22 } catch (Exception e) {
23     e.printStackTrace();
24 }
```

## Streaming input from the file

```
26 char[] cc = new char[100];
27 try(InputStreamReader ipsr2 = new InputStreamReader(new FileInputStream("data.txt"), "UTF-8")){
28     while(ipsr2.read(cc) != -1){
29         System.out.println(new String(cc));
30     }
31 } catch (Exception e){
32     e.printStackTrace();
33 }
```

# Input/Scanner Class

รูปแบบ Scanner

```
Scanner sn = new Scanner(System.in);
```

โดยที่ sn = Object ของ Class Scanner

System.in การอ่านข้อมูลจากแป้นพิมพ์

Method Scanner

- |                 |                            |
|-----------------|----------------------------|
| 1. nextInt()    | รับข้อมูลจำนวนเต็ม         |
| 2. nextFloat()  | รับข้อมูลทศนิยมชนิด float  |
| 3. nextDouble() | รับข้อมูลทศนิยมชนิด double |
| 4. nextLine()   | รับข้อมูลประเภทข้อความ     |

# Input/Scanner Class

Ex1

```
1 import java.util.*;
2
3 public class Study00Scanner
4 {
5     public static void main(String[] args)
6     {
7         String name;
8         Scanner sn = new Scanner(System.in);
9
10        System.out.print("Input Name: ");
11        name = sn.nextLine();
12        System.out.println("");
13        System.out.print("Name : " + name);
14    }
15 }
```

Console

```
<terminated> Study00Scanner [Java A
Input Name: Ko Jirawat
```

```
Name : Ko Jirawat
```

Ex2

```
/// Work2 Input Integer
int id;
Scanner sn_id = new Scanner(System.in);

System.out.print("Input ID: ");
id = sn_id.nextInt();

System.out.print("ID : " + id);
```

Console

```
<terminated> Study00Scanner [
Input ID: 5013139
ID : 5013139
```

# Format Class

ใช้ในการจัดรูปแบบการแสดงผล

รูปแบบ

```
DecimalFormat df = new DecimalFormat(argument);  
String str= df.format(payment);
```

โดยที่ argument เป็นรูปแบบการแสดงผลที่ต้องการ ประกอบด้วย  
0 แทนตัวเลข 0 ในกรณีที่ต้องการให้แสดงตัวเลข  
“#” แทนตัวเลขใดๆ ที่ไม่ใช่ 0 ในกรณีที่เป็นเลข 0 จะไม่แสดงผล  
“,” ให้แสดงผลโดยมีเครื่องหมาย “,” เป็นตัวคั่นในหลักพัน



# Format Class

Ex. 

```
public static void main(String[] args)
{
    int num1 = 1234567;
    DecimalFormat temp1 = new DecimalFormat("#,###");
    System.out.println(temp1.format(num1));

    int num2 = 1234567;
    DecimalFormat temp2 = new DecimalFormat("#,###.00");
    System.out.println(temp2.format(num2));

    double num3 = 1234.567;
    DecimalFormat temp3 = new DecimalFormat("0.###");
    System.out.println(temp3.format(num3));
}
```

 Console  Problems

<terminated> Study00Forma

1,234,567

1,234,567.00

1234.567

# Arrays

# Array creation

- Array can be created using an expression:

`new type[num]`

- An array of *num* elements of type *type* is created
- All elements are initialized to 0, 0.0, false, or null

# Array access

- Let's say we have:

```
int[] a = new int[n];
```

- `a.length` is the length of `a` (which is `n`)
- `a[i]` denotes element number `i` of the array `a` (counting from 0)

# Arrays

รูปแบบ Arrays

```
Datatype[] arrayName;
```

โดยที่ Datatype      เป็น Datatype ของตัวแปรอาร์เรย์ที่ต้องการ  
arrayName      เป็นชื่อตัวแปรอาร์เรย์ ซึ่งตั้งตามกฎการตั้งชื่อ

\*\*\* ต้องกำหนดขนาดของ Arrays ด้วย รูปแบบการกำหนดดังนี้

```
arrayName = new Datatype[n];
```

โดยที่      Datatype      เป็น Datatype ของตัวแปรอาร์เรย์ที่ต้องการ  
arrayName      เป็นชื่อตัวแปรอาร์เรย์ ซึ่งตั้งตามกฎการตั้งชื่อ  
n      เป็นขนาดของตัวแปรอาร์เรย์ที่ต้องการ

หมายเหตุ: การกำหนดสมาชิกให้กับตัวแปรอาร์เรย์ ถือเป็นวิธีการหนึ่งในการกำหนดขนาดของอาร์เรย์

# Array initializers

- Either form is correct:

```
int[] x = { 1, 2, 3, 4, 5 };
```

```
int[] x = new int[] { 1, 2, 3, 4, 5 };
```

# Arrays

Ex. สร้างตัวแปรมาเพื่อเก็บข้อมูล เกรดวิชา Fund2

fund2\_grade ตัวแปรข้อมูลเกรดของนักศึกษา 82 คน

```
/// Work1  
char[] fund2_grade;  
fund2_grade = new char[82];
```

```
/// Work2  
char[] fund2_grade = new char[82];
```

การกำหนดค่า

1. กำหนดพร้อมกับการประกาศตัวแปร
2. กำหนดเองทีหลังจากการสร้างตัวแปร

```
/// Work2  
char[] fund2_grade = {'A', 'B', 'C', 'D'};
```

```
/// Work2  
char[] fund2_grade = new char[82];  
fund2_grade[0] = 'A';  
fund2_grade[1] = 'B';  
fund2_grade[2] = 'C';  
fund2_grade[3] = 'D';
```

```
1  class DieTheDice {
2      public static void main(String[] args) {
3          int[] freq = new int[6];
4          for(int i=0;i<freq.length;i++){
5              int die = (int)(1+6*Math.random()); //Math.random return 0.00 -0.99
6              freq[die-1] += 1;
7          }
8          for(int c=1;c<=6;c++){
9              System.out.println(c+" came up "+freq[c-1]+ " times");
10         }
11     }
12 }
```



# The utility class Arrays

- Package `java.util.Arrays` provides static utility methods for many array operations
- Examples:

```
Arrays.sort(a);
```

```
Arrays.fill(a, v);
```

```
if Arrays.equals(a1, a2) doSomething();
```

# Arrays Methods

มีให้เลือกนำมาออกแบบโปรแกรมมากมาย ดังนี้

1. length      หาจำนวนสมาชิกของ Arrays

```
ArrayName.length
```

2. sort()      เรียงลำดับข้อมูล น้อย>มาก

```
Arrays.sort(ArrayName);
```

3. binarySearch()      ค้นหาข้อมูลใน Arrays

```
indexValue = Arrays.binarySearch(ArrayName, value);
```

4. fill()      ใช้กำหนดค่าข้อมูลให้กับตัวแปรอาร์เรย์

```
Arrays.fill(ArrayName, value);
```

5. equals()      ใช้เปรียบเทียบค่าข้อมูลในตัวแปรอาร์เรย์ (True/False)

# Arrays Methods

Ex. กำหนดให้ `int[] scores = { 13, 22, 27, 18, 21, 26, 11, 8, 29 };`

1. 

```
// Work1 length
int length;
length = scores.length;
System.out.println("length :" +length);
```

Console

<terminated> Stu  
length :9

2. 

```
// Work2 sort()
Arrays.sort(scores);
for(int i = 0; i<scores.length; i++)
{
    System.out.print(scores[i] + ", ");
}
```

Console

<terminated> Study03ArraysMethod [Java Application] C  
8, 11, 13, 18, 21, 22, 26, 27, 29,

# Arrays Methods

Ex. กำหนดให้ `int[] scores = { 13, 22, 27, 18, 21, 26, 11, 8, 29 };`

3. 

```
// Work3 binarySearch()
for(int i = 0; i<scores.length; i++)
{
    System.out.print(scores[i] + ", ");
}
int idx = Arrays.binarySearch(scores, 29);
System.out.println("");
System.out.println("index :" +idx);
```

Console

```
<terminated> Study03ArraysMethod [Java Application]
13, 22, 27, 18, 21, 26, 11, 8, 29,
index :8
```

4. 

```
// Work4 fill()
for(int i = 0; i<scores.length; i++)
{
    System.out.print(scores[i] + ", ");
}
Arrays.fill(scores, 0);
System.out.println("");
for(int i = 0; i<scores.length; i++)
{
    System.out.print(scores[i] + ", ");
}
```

Console

```
<terminated> Study03ArraysMethod [Java Application]
13, 22, 27, 18, 21, 26, 11, 8, 29,
0, 0, 0, 0, 0, 0, 0, 0, 0,
```

# Mutidimensional Arrays

รูปแบบ

```
Datatype[ ][ ] ArrayName;
```

โดยที่	Datatype	เป็นชนิดข้อมูลของตัวแปรอาร์เรย์ที่ต้องการ
	ArrayName	เป็นชื่อตัวแปรอาร์เรย์

ต้องกำหนดขนาดของอาร์เรย์ที่ต้องการก่อนนำไปใช้งานในโปรแกรม ดังนี้

```
ArrayName = new Datatype[m][n];
```

โดยที่	Datatype	เป็นชนิดข้อมูลของตัวแปรอาร์เรย์ที่ต้องการ
	ArrayName	เป็นชื่อตัวแปรอาร์เรย์
	m	เป็นขนาดแถวของตัวแปรอาร์เรย์ที่ต้องการ
	n	เป็นขนาดคอลัมน์ของตัวแปรอาร์เรย์ที่ต้องการ

An example of  
Muti-dimensional  
Arrays  
and it's  
manipulation  
methods

```
5      double[][] r1 = new double[3][2];
6      double[][] r2 = new double[3][];
7      for (int i = 0; i < 3; i++) {
8          r2[i] = new double[2];
9      }
10     if (Arrays.deepEquals(r1, r2)) {
11         System.out.println("Yes, they are equal");
12     } else {
13         System.out.println("No, they are not equal");
14     }
15     double[][] t1 = new double[3][];
16     for (int i = 0; i < 3; i++) {
17         t1[i] = new double[i + 1];
18     }
19     System.out.printf("%s\n", Arrays.deepToString(t1));
20
21     double [][] r3 = {{1.1},{2.2,3.3},{4.4,5.5,6.6}};
22     double [][] r4;
23     r4 = new double[][] {{1.1},{2.2,3.3},{4.4,5.5,6.6}};
24     if (Arrays.deepEquals(r3, r4)) {
25         System.out.println("r3 - r4, they are equal");
26     } else {
27         System.out.println("r3 - r4, they are not equal");
28     }
29     }
```

# Strings

# Strings

- Strings เป็น Datatype? ใน Week1 ค่อนข้างใหม่
- String เป็น Class? มีการสร้าง Object มาเพื่อรับข้อความ

```
2 public class Study05String1
3 {
4     public static void main(String[] args)
5     {
6         String str_id = new String("5013139");
7         System.out.println("ID : " + str_id);
8     }
9 }
```



- String เป็นชนิดข้อมูลที่ประกอบด้วย Char หลาย ๆ ตัวมาเรียงต่อกัน
- เป็นการจองเพื่อเก็บค่าของตำแหน่งอ้างอิงที่ใช้ในการเข้าถึงค่าของข้อมูล  
[ ไม่ใช่จองเพื่อเก็บค่าของข้อมูล ]



# Strings

- การเรียกใช้งาน String นิยมอยู่ 2 รูปแบบ
  1. เรียกใช้แบบการสร้าง Object
  2. เรียกใช้แบบคล้าย Datatypes (มักเจอในรูปแบบนี้)

```
2 public class Study05String1
3 {
4     public static void main(String[] args)
5     {
6         /// Work1
7         String str_id = new String("5013139");
8         System.out.println("ID : " + str_id);
9
10        // Work2
11        String str_name = "Ko Jirawat";
12        System.out.println("Name : " + str_name);
13    }
```

 Console 

```
<terminated> Study05Stri
ID : 5013139
Name :Ko Jirawat
```

# String methods

- Getting its length:
  - `s1.length()`
- Testing for equality:
  - `s1.equals(s2)`
- Retrieving a character at position i:
  - `s1.charAt(i)`
- Converting to a string:
  - `String.valueOf(v)`

# String methods

1. length()      ใช้นับจำนวนตัวอักษร(Char) ใน Strings

รูปแบบ

```
n = str.length();
```

โดยที่	str	ข้อมูลString ที่ต้องการนับจำนวนตัวอักษร(Char)
	n	จำนวนตัวอักษรใน String (Int)

EX

```
// Work1
String str_name = "Ko Jirawat";
System.out.println("Name :" + str_name);

int num_idx;
num_idx = str_name.length();
System.out.println("Index :" + num_idx);
```

Console

```
<terminated> Study06StringM
Name :Ko Jirawat
Index :10
```

# String methods

2. equals() ใช้เปรียบเทียบค่าข้อมูลชนิด String 2 ค่า

รูปแบบ

```
result = str1.equals(str2);
```

โดยที่ str1 ข้อมูล String ชุดที่ 1

str2 ข้อมูล String ชุดที่ 2

result เป็นตัวแปรแบบ Boolean (True/False)

EX

```
// Work2 equal()
String str_id = "5013139";
String str_tmp = "5013149";
boolean result;
result = str_id.equals(str_tmp);

if(result)
{
    System.out.println("Correct id");
}
else
{
    System.out.println("Incorrect id");
}
```

Console

```
<terminated> Study06Str
Incorrect id
```

# String methods

3. charAt() ใช้ค้นหาและแสดงค่าตัวอักษร ที่ตำแหน่งที่ต้องการ

รูปแบบ

```
ch = str.charAt(index);
```

โดยที่	str	String ที่ต้องการค้นหาและแสดงค่าตัวอักษร (Char)
	index	เป็นตำแหน่งของข้อความที่ต้องการค้นหาตัวอักษร
	ch	Datatype char ที่รับค่าตัวอักษรจาก Method charAt()

EX

```
// Work3 charAt()  
String str_name3 = "Ko Jirawat";  
char ch_temp;  
int str_idx = 5;  
ch_temp = str_name3.charAt(str_idx);  
System.out.println("String : " + str_name3);  
System.out.println("String index"+str_idx+" : " +ch_temp);
```

```
Console ✕  
<terminated> Study06StringMet  
String : Ko Jirawat  
String index5 : r
```

# String methods

4. ValueOf()      ใช้ในการเปลี่ยนตัวแปรเป็น Strings

สามารถเปลี่ยนจาก Datatype เป็น String ได้หลายแบบดังนี้

```
String valueOf(boolean b) : String - String
String valueOf(char c) : String - String
String valueOf(char[] data) : String - String
String valueOf(double d) : String - String
String valueOf(float f) : String - String
String valueOf(int i) : String - String
String valueOf(long l) : String - String
String valueOf(Object obj) : String - String
String valueOf(char[] data, int offset, int count) : String - String
```

EX

```
// Work4 valueOf()
char[] arrays_test = {'T','E','S','T'};
char[] arrays_name = {'K','o',' ','J','i','r','a','w','a','t'};
System.out.println("Return Value : " + String.valueOf(arrays_test) );
System.out.println("Return Value : " + String.valueOf(arrays_name) );
```

Console

```
<terminated> Study06StringMethod [Java]
Return Value : TEST
Return Value : Ko Jirawat
```

# String methods

- Concatenating two strings:
  - `s1 + s2`
  - `s1.concat(s2)`
- Comparing lexicographically:
  - `s1.compareTo(s2)`
- Taking a new substring with indexes `i..(j-1)`:
  - `s1.substring(i, j)`

# String methods

5. concat() ใช้สำหรับเชื่อมต่อข้อมูลชนิด String

รูปแบบ

```
str3 = str1.concat(str2);  
str3 = str1+ str2;
```

โดยที่

str1	String ตัวที่ 1
str2	String ตัวที่ 2
str3	String เก็บค่าผลรวมของ str1 กับ str2

EX

```
// Work5 concat()  
String str_1 = "Programming";  
String str_2 = " Fundamental";  
String str_3 = "2";  
String str_4 = " 03603112";  
String str_main;  
str_main = str_1.concat(str_2);  
System.out.println(str_main);  
str_main += (str_3 + str_4);  
System.out.println(str_main);
```

Console

```
<terminated> Study06StringMethod [Java Application]  
Programming Fundamental  
Programming Fundamental2 03603112
```



# String methods

6. compareTo() ใช้เปรียบเทียบค่าข้อมูลชนิด String 2 ค่า

รูปแบบ

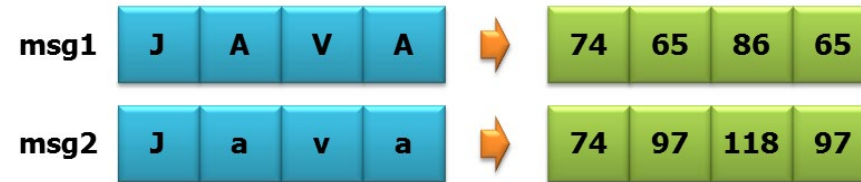
```
n = str1.compareTo(str2);
```

โดยที่

str1	String ตัวที่ 1
str2	String ตัวที่ 2
n	Datatype int เก็บค่าผลที่ได้จากการเปรียบเทียบข้อมูล

EX

```
// Work6 compareTo
String str_1 = "Programming";
String str_2 = " Fundamental";
String str_3 = "Programming";
int compare;
compare = str_1.compareTo(str_2);
System.out.println("1. Compare 1&2 : "+ compare);
compare = str_1.compareTo(str_3);
System.out.println("2. Compare 1&3 : "+ compare);
```



```
Console
<terminated> Study06StringMe
1. Compare 1&2 : 48
2. Compare 1&3 : 0
```

Behide the compareTo()

Decimal	Hexadecimal	Binary	Octal	Char	Decimal	Hexadecimal	Binary	Octal	Char
48	30	110000	60	0	96	60	1100000	140	`
49	31	110001	61	1	97	61	1100001	141	a
50	32	110010	62	2	98	62	1100010	142	b
51	33	110011	63	3	99	63	1100011	143	c
52	34	110100	64	4	100	64	1100100	144	d
53	35	110101	65	5	101	65	1100101	145	e
54	36	110110	66	6	102	66	1100110	146	f
55	37	110111	67	7	103	67	1100111	147	g
56	38	111000	70	8	104	68	1101000	150	h
57	39	111001	71	9	105	69	1101001	151	i
58	3A	111010	72	:	106	6A	1101010	152	j
59	3B	111011	73	;	107	6B	1101011	153	k
60	3C	111100	74	<	108	6C	1101100	154	l
61	3D	111101	75	=	109	6D	1101101	155	m
62	3E	111110	76	>	110	6E	1101110	156	n
63	3F	111111	77	?	111	6F	1101111	157	o
64	40	1000000	100	@	112	70	1110000	160	p
65	41	1000001	101	A	113	71	1110001	161	q

```
String a = new String("JAVa");  
String b = new String("JAVA");  
int compare;  
compare = a.compareTo(b);  
System.out.println(compare);  
//Output = 32
```

$$97 - 65 = ?$$

# String methods

7. substring()                      ใช้สำหรับตัดข้อมูลชนิด String ให้เป็นข้อความย่อย

รูปแบบ

```
str2 = str1.substring(x, y);
```

โดยที่	str1	String ที่ต้องการตัด
	str2	String ที่รับค่าข้อความที่ได้จากเมธอด substring()
	x	ตำแหน่ง Start ตัดข้อความ
	y	ตำแหน่ง Stop ที่ตัดข้อความ

EX

```
// Work7 substring
String str_1 = "Programming Fundamental2 03603112";
String str_cut;
str_cut = str_1.substring(12, 24);
System.out.println(str_1);
System.out.println(str_cut);
```

Console

```
<terminated> Study06StringMethod [Java Application]
Programming Fundamental2 03603112
Fundamental2
```

# ArrayLists

# ArrayList

- An ArrayList is closest to a Python list
- Normal arrays are fixed-length
- ArrayList is dynamic
- Declaration and initialization:

```
ArrayList<Integer> a = new ArrayList<Integer>();
```

# ArrayLists

- เป็นโครงสร้างที่ใช้ Arrays จัดเก็บข้อมูลเป็นลำดับที่ต่อเนื่อง
- อ้างถึงข้อมูลในลำดับ ด้วยการระบุตำแหน่ง (index)
- รองรับการเก็บข้อมูลแบบ Object ซึ่งเป็น Arrays ของ Reference
- [ สามารถขยายขนาดได้เองโดยอัตโนมัติ ] ในขณะที่อาร์เรย์ไม่สามารถเปลี่ยนแปลงขนาดของอาร์เรย์หลังจากที่ถูกสร้างขึ้นได้

Ex. 

```
ArrayList<Integer> my_grad1 = new ArrayList<Integer>();  
ArrayList my_grad2 = new ArrayList();
```

# ArrayList methods

- For ArrayList<T>
  - boolean add(T o)
  - void add(int i, T o)
  - T get(int i)
  - T remove(int i)
  - T set(int i, T o)
  - ArrayList<T> subList(int from, int to)

# ArrayList methods

1. add()            เพิ่มสมาชิกใน ArrayList

รูปแบบ

```
arrayListName.add(objectValue);
```

โดยที่    arrayListName    Object ArrayList ที่ต้องการเพิ่มสมาชิก  
          objectValue       ข้อมูลที่ต้องการเพิ่มใน ArrayList

สามารถเพิ่มสมาชิกที่ตำแหน่งใดๆได้

รูปแบบ

```
arrayListName.add(index, objectValue);
```

โดยที่    index            เป็นตำแหน่งที่ต้องการเพิ่มสมาชิก



# ArrayList methods

## 1. add()            เพิ่มสมาชิกใน ArrayList

```
// Work2 add()
ArrayList<Integer> my_num = new ArrayList<Integer>();
for(int i = 0 ; i<= 5; i++)
{
    my_num.add(i);
}
System.out.println(my_num);

my_num.add(10);                // Add Value 10 to last index
System.out.println(my_num);

my_num.add(1,11);              // Add Value 11 to 1 index
System.out.println(my_num);
```

Console ✕

<terminated> Study07ArrayList [Java Appli

[0, 1, 2, 3, 4, 5]

[0, 1, 2, 3, 4, 5, 10]

[0, 11, 1, 2, 3, 4, 5, 10]

# ArrayList methods

2. get()                    หาค่าสมาชิกใน ArrayList ณ ตำแหน่งที่ต้องการ

รูปแบบ

objectValue = arrayListName.get(index);

โดยที่	arrayListName	Object ArrayList ที่ต้องการหาค่าสมาชิก
	objectValue	เป็นออบเจกต์ที่ใช้รับข้อมูลจาก Method
	index	เป็นตำแหน่งที่ต้องการหาค่าสมาชิก

EX

```
// Work3 get()
ArrayList<Integer> my_num = new ArrayList<Integer>();
for(int i = 0 ; i<= 5; i++)
{
    my_num.add(i+10);
}
System.out.println(my_num);            // Print all value
System.out.println("Object index1 :" + my_num.get(1));
```

```
Console ✕
<terminated> Study07ArrayList [Java A
[10, 11, 12, 13, 14, 15]
Object index1 :11
```

# ArrayList methods

3. remove() สำหรับลบสมาชิก ณ ตำแหน่งที่ต้องการออกจาก ArrayList

รูปแบบ

```
arrayListName.remove(index);
```

โดยที่     arrayListName     Object ArrayList ที่ต้องการลบสมาชิก  
             index                เป็นตำแหน่งที่ต้องการลบ

EX

```
// Work4 remove()
ArrayList<Integer> my_num = new ArrayList<Integer>();
for(int i = 0 ; i<= 5; i++)
{
    my_num.add(i+10);
}
System.out.println(my_num);      // Print all value
my_num.remove(3);
System.out.println(my_num);      // Print all value
```

Console

```
<terminated> Study07ArrayList [Java A
[10, 11, 12, 13, 14, 15]
[10, 11, 12, 14, 15]
```

# ArrayList methods

4. set()                      สำหรับแก้ไขค่าสมาชิก ณ ตำแหน่งที่ต้องใน ArrayList

รูปแบบ

arrayListName.set(index, objectValue);

โดยที่	arrayListName	Object ArrayList ที่ต้องการแก้ไขสมาชิก
	index	เป็นตำแหน่งที่ต้องการลบ
	objectValue	ข้อมูลที่ต้องการแก้ไขใน ArrayList

EX

```
// Work4 set()
ArrayList<Integer> my_num = new ArrayList<Integer>();
for(int i = 0 ; i<= 5; i++)
{
    my_num.add(i+10);
}
System.out.println(my_num);           // Print all value
my_num.set(3, null);
System.out.println(my_num);           // Print all value
```

Console

```
<terminated> Study07ArrayList [Java Appli
[10, 11, 12, 13, 14, 15]
[10, 11, 12, null, 14, 15]
```

# Looping over a collection

- Form:

```
for (type x : expression)  
    body
```

- Its behaviour is very similar to the Python's for loop
- Example:

```
int[] iarr = { 2, 3, 5, 7, 11 };  
int sum = 0;  
for (int i : iarr)  
    sum += i;  
System.out.println(sum);
```

# Looping over a collection

- คำสั่ง วนลูปเพื่อเข้าถึงข้อมูลแต่ละตำแหน่งใน Array หรือ ArrayList
- ป้องกันการอ้างถึงข้อมูลที่เกิดขอบเขต (IndexOutOfBoundsException)
- วนลูปจะเริ่มต้นตั้งแต่ตำแหน่งแรกจนถึงตำแหน่งสุดท้าย ไม่สามารถกำหนดขอบเขตได้

รูปแบบ

```
for (type x : expression)
{
    Body;
}
```

โดยที่

type

ข้อมูล Array หรือ ArrayList

x

เป็นชื่อตัวแปรที่ใช้เข้าถึงข้อมูลในแต่ละรอบ

expression

ชื่อตัวแปร Array หรือ ArrayList

Body

เป็นชุดของคำสั่งที่ต้องการให้ทำงานในแต่ละรอบ

# Looping over a collection

EX1

int

```
// Work 1
int[] iarr = { 2, 3, 5, 7, 11 };
int sum = 0;

for(int i : iarr)
{
    sum += i;
}
System.out.println("sum : " + sum);
```

```
Console
<terminated> Study08
sum : 28
```

EX2

double

```
// Work 2
double[] scores_midterm = { 13.0, 22.0, 27.0, 18.0, 21.0, 26.0, 11.0, 8.0, 29.0 };

for(double calculate : scores_midterm)
{
    System.out.println("Scores: " + calculate + " is percent: " + (calculate/30*100));
}
```

```
Console
<terminated> Study08ForEachLoop [Java Application] C:\Program F
Scores: 13.0 is percent: 43.333333333333336
Scores: 22.0 is percent: 73.33333333333333
Scores: 27.0 is percent: 90.0
Scores: 18.0 is percent: 60.0
Scores: 21.0 is percent: 70.0
Scores: 26.0 is percent: 86.66666666666667
Scores: 11.0 is percent: 36.666666666666664
Scores: 8.0 is percent: 26.666666666666668
Scores: 29.0 is percent: 96.66666666666667
```

enum



# enum

เกิดจากข้อจำกัดของค่าคงที่ Datatype พื้นฐาน

Ex. การประกาศค่าคงที่ในการเก็บเกรด วิชา Fund2

```
final static char grade_A = 'A';  
final static char grade_B = 'B';  
final static char grade_C = 'C';  
final static char grade_D = 'D';  
final static char grade_F = 'F';
```

\*\* ซึ่งตัวแปรไม่มีความจำเป็นต้องนำค่าคงที่จากเกรดทั้ง5มาใส่

\*\* ข้อมูลพื้นฐาน ที่ไม่มี Method ที่นำมาคำนวณได้

```
char ch = grade_A + grade_F;  
System.out.println(ch);
```

## enum คือ?

1. เป็น ชนิดของข้อมูล ที่สร้างขึ้นใหม่ด้วยตัวเอง
2. enum จะถูกนำมาใช้เมื่อต้องการกำหนดค่าทั้งหมดที่เป็นไปได้ให้กับตัวแปร
3. ใช้หลักการตั้งชื่อแบบค่าคงที่ คือ ตัวพิมพ์ใหญ่ คั่นด้วย \_ (underscore)

รูปแบบ

```
enumName { value1, value2,..., valueN }
```

โดยที่

enumName

ตัวแปร enum โดยตั้งตามกฎการตั้งชื่อ

value1, value2, ..., valueN

เป็นค่าข้อมูลที่จะกำหนดให้ตัวแปร enum

EX. enum

- |                 |       |                                     |
|-----------------|-------|-------------------------------------|
| 1. เกรด         | 8 ค่า | [ A, B+, B, C+, C, D+, D, F ]       |
| 2. เพศ          | 2 ค่า | [ ชาย, หญิง ]                       |
| 3. ไฟจราจร      | 3 ค่า | [ แดง, เหลือง, เขียว ]              |
| 4. วันในสัปดาห์ | 7 ค่า | [ mon, tu, wed, th, fri, sat, sun ] |

## EX. Enum

กำหนด enum เกรด

```
enum Grade {A, BP, B, CP, C, DP, D, F}
```

```
// Work1  
Grade myGrade= Grade.A;  
System.out.println("My Grade : "+myGrade);
```

Console Problems

<terminated> Study09Enum2 [Ja  
My Grade : A



```
// Work2  
Grade myGrade = Grade.A;  
myGrade = 'I';  
myGrade= Grade.I;  
System.out.println(myGrade);
```

# HashMaps

# HashMap

- A HashMap is similar to a Python dictionary

## EX. HashMap

```
HashMap<String, Integer> wdayNumber = new HashMap<String, Integer>();

int wdayNo = 1;
String day = "Tuesday";

String[] wdays = { "Monday", "Tuesday", "Wednesday", "Thursday", "Friday", "Saturday", "Sunday"};

for (String wday : wdays)
{
    wdayNumber.put(wday, wdayNo++);
}

Integer dayOfWeek = wdayNumber.get(day);

if (dayOfWeek == null)
{
    System.out.println("Not a day name");
}
else
{
    System.out.println(day + " Day of Week: " + dayOfWeek);
    System.out.println(wdayNumber);
}
```

Console Problems

<terminated> Study10HashMap [Java Application] C:\Program Files\Java\jre1.8.0\_121\bin\javaw.exe (27 м.к. 2560 22:29:56)

Tuesday Day of Week: 2

{Monday=1, Thursday=4, Friday=5, Sunday=7, Wednesday=3, Tuesday=2, Saturday=6}