



Bài 2. Cú pháp Java cơ bản

Lý thuyết và ngôn ngữ hướng đối tượng
(bài tập)



Bài 1

- Viết một chương trình có tên là Arithmetic thực hiện tính tổng và giá trị trung bình các số nguyên trong một mảng 1 chiều được cho sẵn

- ```
Array<Integer> data =
 {78, 65, 78, 21, 93, 45, 33, 55, 22, 81} ;
```

- Kết quả mong đợi khi chạy chương trình:

```
java Arithmetic
sum: 571
avg: 57.1
```



# Bài 1

---

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

 int data[] = {
 78, 65, 78, 21, 93, 45, 33, 55
 , 22, 81 };
 int sum = 0;
 // Total value
 for(int i=0;
 i<data.length; i++){
 // Calculation of total
 sum += data[i];
 }
}
```

```
/* Use extension loop 'for'
for(int num : data){
 sum += data;
}*/

float ave = sum/10.0f;
// (Another solution)
float ave =
 (float)sum/10;
System.out.println("Sum
:" + sum);
System.out.println("Ave
:" + ave);
}
}
```



# Bài 1

---

- Tiếp tục bài tập trước để chương trình hiển thị giá trị Min, Max của mảng.



## Bài 2

---

- Viết một chương trình tính tổng sau, với giá trị  $n$  là số nguyên dương nhập vào từ bàn phím:

$$\textit{Harmonic}(n) = 1 + \frac{1}{2} + \frac{1}{3} + \dots + \frac{1}{n}$$



# Bài 3 - Javadoc

---

- Công cụ lấy ra lớp và phương thức, v.v., được xác định trong tệp nguồn và tạo hướng dẫn tham khảo của chương trình
- Thẻ chính của `javadoc`,
  - `@see` tên tham chiếu: Tạo liên kết tham chiếu của các lớp khác và các gói liên quan từ lớp của đối tượng.
  - `@exception` giải thích tên lớp ngoại lệ: Mô tả giải thích về ngoại lệ mà phương thức của đối tượng có khả năng ném.
  - `@param` giải thích tên đối số
  - `@return` giải thích về giá trị trả về của phương thức của đối tượng được mô tả.
  - ...



# Circle.java

```
import java.awt.*;
import java.applet.*;
```

```
/**
 * A simple circle class.
 *
 * @author Samuel A. Rebelsky
 * @version 1.1 of February 1999
 */
public class Circle {
```

```
 /** The current color of the circle. */
 protected Color color;
```

```
 /** The current diameter of the circle. */
 protected int diameter;
```

```
 /**
```

```
 * Create a new circle with specified
 diameter and color.
```

```
 */
```

```
 public Circle(int diameter, Color color) {
 this.diameter = diameter;
 this.setColor(color);
 }
```

```
 /**
```

```
 * Create a new circle with specified
 diameter and default color.
```

```
 */
```

```
 public Circle(int diameter) {
 this.diameter = diameter;
 this.setColor(Color.blue); // Blue is as
 good a default as any.
 }
```



# Circle.java

```
/**
 * Get the diameter of this circle.
 */
public int getSize() {
 return this.diameter;
}
```

```
/**
 * Set the color of this circle.
 */
public void setColor(Color newColor) {
 this.color = newColor;
}

/**
 * Set the diameter of this circle.
 */
public void setSize(int newDiameter) {
 this.diameter = newDiameter;
}
}
```



# Result

## Class Circle

java.lang.Object  
└ Circle

```
public class Circle
extends java.lang.Object
```

A simple circle class.

## Field Summary

|                             |                                                                 |
|-----------------------------|-----------------------------------------------------------------|
| protected<br>java.awt.Color | <a href="#">color</a><br>The current color of the circle.       |
| protected int               | <a href="#">diameter</a><br>The current diameter of the circle. |

## Constructor Summary

|                                                                                                                       |
|-----------------------------------------------------------------------------------------------------------------------|
| <a href="#">Circle</a> (int diameter)<br>Create a new circle with specified diameter and default color.               |
| <a href="#">Circle</a> (int diameter, java.awt.Color color)<br>Create a new circle with specified diameter and color. |

## Method Summary





# Bài 4

- Bài kiểm tra Kỹ sư Công nghệ Thông tin được thực hiện ba lần trong một công ty
  - Các thí sinh có năm người (Okada, Tamura, Matsuda, Asai và Natsuki)
  - Điểm số là Okada {79, 65, 78}, Tamura {21, 93, 45}, Matsuda {31, 55, 22}, Asai { 81, 66, 81} và Natsuki {76, 90, 86}.
- Kết quả kiểm tra và tên của người kiểm tra được lưu trữ trong mảng như sau.  

```
String name[] = {"Okada", "Tamura", "Matsuda",
"Asai", "Natsuki"};
int data[][] = {{79, 65, 78}, {21, 93, 45}, {31, 55,
22}, {81, 66, 81}, {76, 90, 86}};
```
- Viết chương trình tính điểm trung bình của mỗi người kiểm tra từ mảng này và tạo một biểu đồ thanh cho thấy mức độ thành tích của họ.
  - Biểu đồ thanh hiển thị thành tích theo số '\*' và ghi một '\*' cho 10 điểm của điểm trung bình.



# Solution

```
class TestResult {
public static void main(String args[]) {
 String name[] =
 {"Okada","Tamura","Matsuda","Asai",
 "Natsuki"}; // Name
 // Score of 3 courses
 int data[][] =
 {{79,65,78},{21,93,45},{31,55,22},
 {81,66,81},{76,90,86}};
 // Array that stores total points
 int sum[] = new int[5];
 // Calculation of total
 for(int i=0;i<5;i++) {
 for(int j=0;j<3;j++) {
 sum[i] += data[i][j];
 }
 }
}
```

```
for(int i=0;i<5;i++) {
 // '*'Calculation of number
 int stars = (sum[i] / 3) / 10;
 // Display of names
 System.out.print("Name:" + name[i] + " ");
 // Display of average score
 System.out.print("Ave :" + sum[i]/3.0f + " ");
 // Display of achievement degree
 for(int j=0;j<stars;j++){ {
 System.out.print('*');
 }
 System.out.println();
}
}
}
```