

Bài 2

Các khái niệm cơ bản về ngôn ngữ lập trình Java

NỘI DUNG

- Biến - Variables
- Mảng - Arrays
- Kiểu dữ liệu - Data types
- Toán tử - Operators
- Control Flow Statements

References:

<https://docs.oracle.com/javase/tutorial/java/nutsandbolts/index.html>

Quy ước đặt tên biến

- Tên biến có phân biệt chữ hoa chữ thường
- Bắt đầu bằng 1 ký tự chữ, ký tự đô la "\$", hoặc ký tự gạch dưới "_"
- Khoảng trắng không được chấp nhận
- Tên biến phải khác với từ khóa (keywords)

Quy ước đặt tên biến

- Nếu 1 từ: tất cả chữ thường:
student, teacher,...
- Nếu tên biến nhiều hơn 1 từ thì ký tự đầu của mỗi từ là chữ hoa:
StudentName, TeacherName,...
- Tên hằng thì viết hoa hết:
NUM_1, NUM_2,...

Primitive Data Types (Các kiểu dữ liệu nguyên thủy)

- Kiểu dữ liệu nguyên thủy không phải là kiểu dữ liệu HĐT mà là một kiểu dữ liệu đơn (chứa 1 giá trị)

Data Type	Bytes	Default Value (for fields)
byte	1	0
short	2	0
int	4	0
long	8	0L
float	4	0.0f
double	8	0.0d
char	2	'\u0000' (or 0)
boolean	true/false	false

String – Object Data Type

- Tạo một chuỗi:
 - String name = "Steve";
 - String s = "";
 - String str = null;
- String concatenation (nối chuỗi)
 - String str = "foo" + "bar" + "!";

Array – Mảng

- Khai báo 1 mảng

int A[]; //A là tên mảng

double B[]; //B là tên mảng

- Cấp phát bộ nhớ cho mảng

A = new int[10];//dùng từ khóa **new**

B = new double[10];

- Khởi tạo giá trị cho phần tử của mảng:

A[0] = 5; //Gán phần tử thứ 0 bằng 5

A[1] = 6; //Gán phần tử thứ 1 bằng 6

- Kích thước của mảng:

System.out.println(anArray.length);

Operators – Toán tử

Operators – Toán tử	Mô tả
Toán tử tăng/giảm	expr++ expr-- hoặc ++expr --expr
Nhân, chia, lấy phần dư	* , / , %
Cộng, trừ	+ , -
Phép toán quan hệ	< , > , <= , >=
So sánh bằng/không bằng	== , !=
And	&&
Or	
Ternary (phép toán 3 ngôi)	expression ? if true : if not true Ví dụ: int result = (x==1) ? 1 : 2;
Phép gán	= , += , -= , *= , /= , %=

Statements and Blocks

(Câu lệnh và khối)

- A *statement* forms a complete unit of execution
- A *block* is a group of zero or more statements between balanced braces

```
// declaration statement
double aValue = 8933.234;
// assignment statement
aValue = 8933.234;
// increment statement
aValue++;
// method invocation statement
System.out.println("Hello World!");
```

Blocks Demo 1

```
class BlockDemo {  
    public static void main(String[] args) {  
        boolean condition = true;  
        if (condition) { // begin block 1  
            System.out.println("Condition is true.");  
        } // end block 1  
        else { // begin block 2  
            System.out.println("Condition is false.");  
        } // end block 2  
    }  
}
```

Blocks Demo 2

```
1
2     public class HelloWorld {
3         public static void main(String[] args) {
4             int n=10,m=0;
5             if (n > 0) {
6                 int tong =0;
7                 tong +=n;
8             }
9             System.out.println(tong);
10
11            for (int i = 0; i < 10; i++)
12                n+=i;
13
14        }
15    }
```

Tại sao
lỗi

Tại sao
lỗi

Các câu lệnh điều khiển

- Các câu lệnh rẽ nhánh:
 - if-then, if-then-else, switch
- Các câu lệnh lặp
 - for, while, do-while
- Các câu lệnh sử dụng trong vòng lặp:
 - break, continue, return

if-else Demo

```
class IfElseDemo {  
    public static void main(String[] args) {  
  
        int testscore = 76;  
        char grade;  
  
        if (testscore >= 90) {  
            grade = 'A';  
        } else if (testscore >= 80) {  
            grade = 'B';  
        } else if (testscore >= 70) {  
            grade = 'C';  
        } else if (testscore >= 60) {  
            grade = 'D';  
        } else {  
            grade = 'F';  
        }  
        System.out.println("Grade = " + grade);  
    }  
}
```

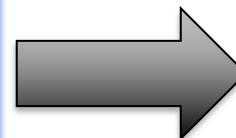
switch Demo

```
int month = 8;
String monthString;
switch (month) {
    case 1: monthString = "January";
    break;
    case 2: monthString = "February";
    break;
    case 3: monthString = "March";
    break;
    case 4: monthString = "April";
    break;
    case 5: monthString = "May";
    break;
    case 6: monthString = "June";
    break;
    case 7: monthString = "July";
    break;
    case 8: monthString = "August";
    break;
    case 9: monthString = "September";
    break;
    case 10: monthString = "October";
    break;
    case 11: monthString = "November";
    break;
    case 12: monthString = "December";
    break;
    default: monthString = "Invalid month";
    break;
}
System.out.println(monthString);
```

while và do-while

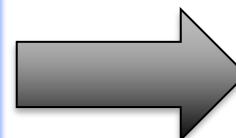
```
1 package helloworld;
2 public class HelloWorld {
3     public static void main(String[] args) {
4         int dem = 1;
5         while (dem < 11)
6         {
7             System.out.println("Dem " + dem);
8             dem++;
9         }
10    }
11 }
```

Kiểm tra đk lặp trước



```
Output - HelloWorld (run) ×
Dem 1
Dem 2
Dem 3
Dem 4
Dem 5
Dem 6
Dem 7
Dem 8
Dem 9
Dem 10
BUILD SUCCESSFUL (total time: 0 seconds)
```

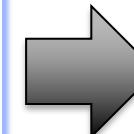
```
1 package helloworld;
2 public class HelloWorld {
3     public static void main(String[] args) {
4         int dem = 1;
5         do
6         {
7             System.out.println("Dem " + dem);
8             dem++;
9         } while (dem < 11);
10    }
11 }
```



Kiểm tra đk lặp sau

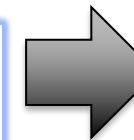
for và for-each

```
1 package helloworld;
2 public class HelloWorld {
3     public static void main(String[] args) {
4         for (int i = 1; i<=10; i++)
5         {
6             System.out.println("Dem " + i);
7         }
8     }
9 }
```



```
Output - HelloWorld [Run] X
Dem 1
Dem 2
Dem 3
Dem 4
Dem 5
Dem 6
Dem 7
Dem 8
Dem 9
Dem 10
BUILD SUCCESSFUL (total time: 0 seconds)
```

```
1 package helloworld;
2 public class HelloWorld {
3     public static void main(String[] args) {
4         int A[] = {1,2,3,4,5,6,7,8,9,10};
5         for (int item: A)
6         {
7             System.out.println("Dem " + item);
8         }
9     }
10 }
```

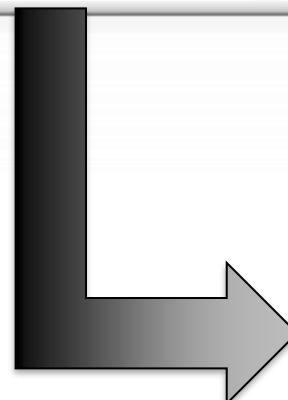


break Demo

```
1 package helloworld;
2 public class HelloWorld {
3     public static void main(String[] args) {
4         int A[] = {1,2,3,4,5,6,7,8,9,10};
5         for (int item: A)
6         {
7             System.out.println("Dem " + item);
8             if (item % 5 == 0)
9                 break;
10        }
11    }
12 }
```

Thoát khỏi
vòng lặp khi
nào?

```
TS
TT
TO
```

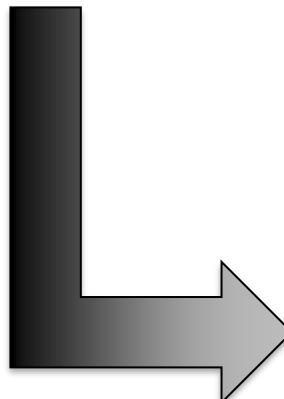


```
run:
Dem 1
Dem 2
Dem 3
Dem 4
Dem 5
BUILD SUCCESSFUL (total time: 0 seconds)
```

continue Demo

```
1 package helloworld;
2 public class HelloWorld {
3     public static void main(String[] args) {
4         int A[] = {1,2,3,4,5,6,7,8,9,10};
5         for (int item: A)
6         {
7             if (item % 2 == 0)
8                 continue;
9             System.out.println("Dem " + item);
10        }
11    }
12 }
```

Lệnh **continue**
làm gì đây?



```
run:
Dem 1
Dem 3
Dem 5
Dem 7
Dem 9
BUILD SUCCESSFUL (total time: 0 seconds)
```

Infinite loop

Vòng lặp vô tận

- Vòng lặp for vô tận

```
for (; ; ) { //code here}
```

- Vòng lặp while vô tận

```
while (true) { //code here}
```

- Vòng lặp do-while vô tận

```
do { //code here} while (true);
```



Sử dụng lệnh gì để
thoát vòng lặp vô
tận?

Questions and Exercises

- Variables:

[https://docs.oracle.com/javase/tutorial/java/nutsandbolts
/QandE/questions_variables.html](https://docs.oracle.com/javase/tutorial/java/nutsandbolts/QandE/questions_variables.html)

- Operators:

[https://docs.oracle.com/javase/tutorial/java/nutsandbolts
/QandE/questions_operators.html](https://docs.oracle.com/javase/tutorial/java/nutsandbolts/QandE/questions_operators.html)

- Expressions, Statements, and Blocks:

[https://docs.oracle.com/javase/tutorial/java/nutsandbolts
/QandE/questions_expressions.html](https://docs.oracle.com/javase/tutorial/java/nutsandbolts/QandE/questions_expressions.html)

- Control Flow Statements:

[https://docs.oracle.com/javase/tutorial/java/nutsandbolts
/QandE/questions_flow.html](https://docs.oracle.com/javase/tutorial/java/nutsandbolts/QandE/questions_flow.html)

HỎI ĐÁP

