



Cấu trúc điều khiển trong Java

CN.TRẦN HẢI LONG

BỘ MÔN CNPM – KHOA CÔNG NGHỆ THÔNG TIN – TRƯỜNG ĐHSPHN

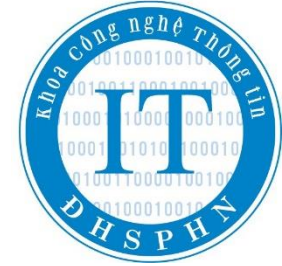
EMAIL: longth@hnue.edu.vn

PHONE: 0966736098

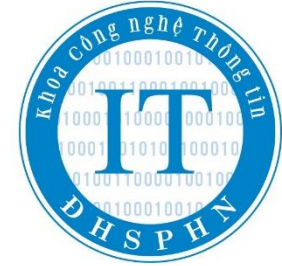


Nội dung

- I. Cấu trúc rẽ nhánh trong Java**
- II. Cấu trúc lặp trong Java**

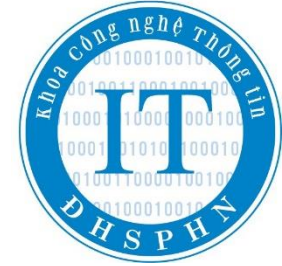


I. Cấu trúc rẽ nhánh trong Java



I. Cấu trúc rẽ nhánh trong Java

- ❖ Dùng để giải quyết các bài toán xử lý công việc đi kèm với mệnh đề điều kiện.
- ❖ Ví dụ:
 - Nếu là thứ 2, mình sẽ học toán.
 - Nếu là thứ 3, mình sẽ học văn.



I. Cấu trúc rẽ nhánh trong Java

1.1 Cấu trúc if...else

1.2 Cấu trúc switch...case

1.3 Một số ví dụ



I. Cấu trúc rẽ nhánh trong Java

1.1 Cấu trúc if...else



I. Cấu trúc rẽ nhánh trong Java

1.1 Cấu trúc if...else

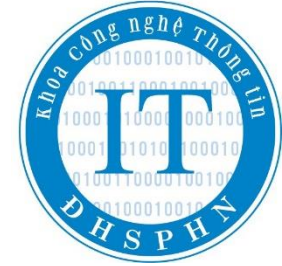
- ❖ Giải quyết công việc nếu thỏa mãn mệnh đề điều kiện.
- ❖ Các mệnh đề điều kiện được xử lý tuần tự.



I. Cấu trúc rẽ nhánh trong Java

1.1 Cấu trúc if...else

Các dạng mệnh đề if...else	Ví dụ
<pre>if(bieu_thuc_dieu_kien){ //Thực hiện công việc }</pre>	<pre>int a = 4; if(a%2 == 0){ System.out.println(a + " la so chan"); }</pre>
<pre>if(bieu_thuc_dieu_kien){ //Công việc 1 } else{ //Công việc 2 }</pre>	<pre>int a = 4; if(a%2 == 0){ System.out.println(a + " la so chan"); } else { System.out.println(a + " khong la so chan"); }</pre>
<pre>if(bieu_thuc_dieu_kien1){ //Công việc 1 } else if(bieu_thuc_dieu_kien2){ //Công việc 2 } else{ //Công việc 3 }</pre>	<pre>int a = 4; if(a < 2){ System.out.println(a + " nho hon 2"); } else if(a == 2){ System.out.println(a + " bang 2"); } else { System.out.println(a + " lon hon 2"); }</pre>



I. Cấu trúc rẽ nhánh trong Java

1.2 Cấu trúc switch...case



I. Cấu trúc rẽ nhánh trong Java

1.2 Cấu trúc switch...case

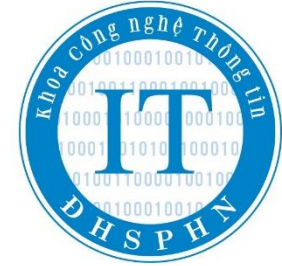
- ❖ Giải quyết công việc nếu thỏa mãn mệnh đề điều kiện.
- ❖ Các mệnh đề điều kiện được xử lý cùng một lúc thông qua cơ chế so sánh **key - value**.
- ❖ **Key**: Là một giá trị hoặc 1 biến.
- ❖ **value1, value2,...** là các hằng số giá trị để so sánh với **key**.
- ❖ **break**: Là lệnh ngắt khỏi switch...case khi công việc thực hiện xong.
- ❖ **default**: Là đại diện cho các trường hợp không xuất hiện trong **case**.



I. Cấu trúc rẽ nhánh trong Java

1.2 Cấu trúc switch...case

Các dạng mệnh đề switch...case	Ví dụ
<pre>switch(key){ case value1: //Thực hiện công việc khi key = value1 break; case value2: //Thực hiện công việc khi key = value2 break; default: //Thực hiện công việc khi key != value1, value2 break; }</pre>	<pre>int thu = 2; switch(thu){ case 2: System.out.println("Hoc Toan"); break; case 3: System.out.println("Hoc Van"); break; default: System.out.println("Duoc nghi"); break; }</pre>
<pre>switch(key){ case value1: case value2: //Thực hiện công việc khi key = value1, value2 break; case value3: //Thực hiện công việc khi key = value3 break; default: //Thực hiện công việc khi key != v1, v2, v3 break; }</pre>	<pre>switch(thu){ case 2: case 4: System.out.println("Hoc Toan"); break; case 3: case 5: System.out.println("Hoc Van"); break; default: System.out.println("Duoc nghi"); break; }</pre>



I. Cấu trúc rẽ nhánh trong Java

1.3 Một số ví dụ



I. Cấu trúc rẽ nhánh trong Java

1.3 Một số ví dụ

Ví dụ 1: Viết chương trình giải phương trình bậc 1: $ax + b = 0$

```
SinhVien.java
1 package com.hnue.myproject.package1;
2 public class SinhVien {
3     public static void main(String[] args) {
4         double a = 4, b = 5.7;
5         if(a == 0 & b == 0)
6         {
7             System.out.println("Phương trình " + a + "x + " + b + " = 0 vô số nghiệm");
8         }
9         else
10        {
11            if(a == 0 && b != 0)
12            {
13                System.out.println("Phương trình " + a + "x + " + b + " = 0 vô nghiệm");
14            }
15            else
16            {
17                System.out.println("Phương trình " + a + "x + " + b + " = 0 có nghiệm x = " + -b/a);
18            }
19        }
20    }
21 }
```

Console

<terminated> SinhVien [Java Application] C:\Users\Hailong\p2\pools\plugins\org.eclipse.justi.openjdk.hotspot.jre.full.win32.x86_64_14.0.2.v20200815-0932\jre\bin\javaw.exe (Oct 28, 2020, 9:42:35 PM - 9:42:35 PM)

Phương trình 4.0x + 5.7 = 0 có nghiệm x = -1.425

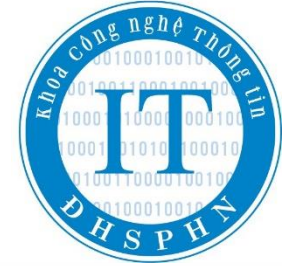


I. Cấu trúc rẽ nhánh trong Java

1.3 Một số ví dụ

Ví dụ 2: Viết chương trình nhập vào 2 số thực x , y và phép toán $+$, $-$, $*$, $/$.

- Nếu là $+$ thì in ra kết quả $x + y$.
- Nếu là $-$ thì in ra kết quả $x - y$.
- Nếu là $*$ thì in ra kết quả $x * y$.
- Nếu là $/$ thì in ra kết quả x / y ($y = 0$ thì thông báo không chia được).



I. Cấu trúc rẽ nhánh trong Java

1.3 Một số ví dụ Ví dụ 2:

```
SinhVien.java
1 package com.hnue.myproject.package1;
2 public class SinhVien {
3     public static void main(String[] args) {
4         double x = 6, y = 4;
5         char phepToan = '/';
6         switch(phepToan)
7         {
8             case '+':
9                 System.out.println(x + " + " + y + " = " + (x + y)); break;
10            case '-':
11                System.out.println(x + " - " + y + " = " + (x - y)); break;
12            case '*':
13                System.out.println(x + " * " + y + " = " + (x * y)); break;
14            case '/':
15                if(y != 0)
16                {
17                    System.out.println(x + " / " + y + " = " + (x / y)); break;
18                }
19                else
20                {
21                    System.out.println("Không chia được vì mẫu số y = 0"); break;
22                }
23        }
24    }
25 }
```

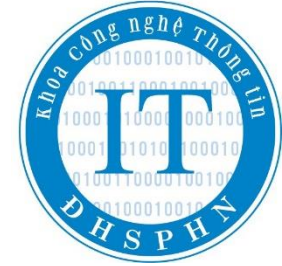
Console

<terminated> SinhVien [Java Application] C:\Users\HaiLong\p2\pool\plugins\org.eclipse.justj.openjdk.hotspot.jre.full.win32.x86_64_14.0.2.v20200815-0932\jre\bin\javaw.exe (Oct 28, 2020, 10:40:14 PM – 10:40:14 PM)

6.0 / 4.0 = 1.5



II. Cấu trúc lặp trong Java



II. Cấu trúc lặp trong Java

- ❖ Dùng để giải quyết các bài toán mà công việc cần lặp đi lặp lại nhiều lần.
- ❖ Ví dụ:
 - Nhập giá trị cho các phần tử trong mảng.
 - Tìm tất cả các số nguyên tố nhỏ hơn 1000.
- ❖ Có 3 loại vòng lặp trong Java:
 - for
 - while
 - do...while



II. Cấu trúc lặp trong Java

2.1 Cấu trúc lặp for

2.2 Cấu trúc lặp while

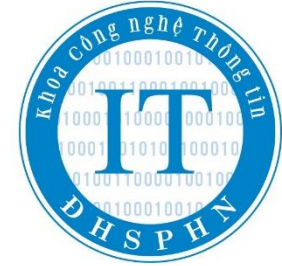
2.3 Cấu trúc lặp do...while

2.4 Các lệnh dừng vòng lặp



II. Cấu trúc lặp trong Java

2.1 Cấu trúc lặp for



II. Cấu trúc lặp trong Java

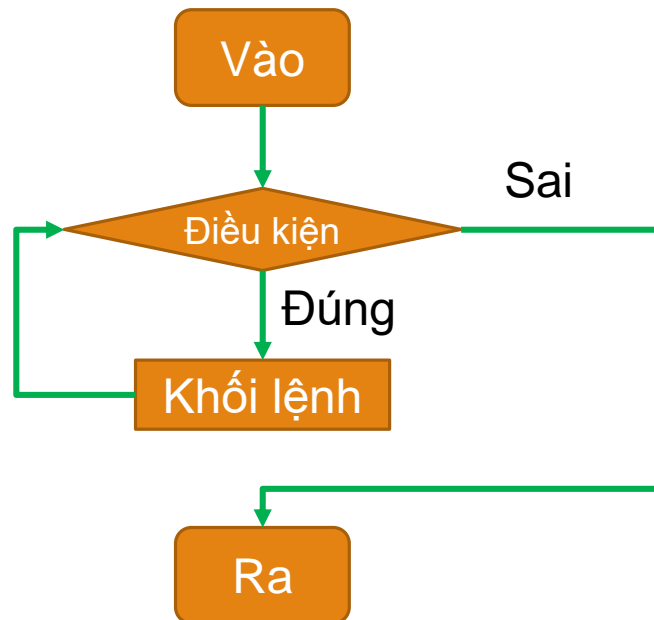
2.1 Cấu trúc lặp for

❖ Thực hiện lặp đi lặp lại công việc với số lần lặp xác định.

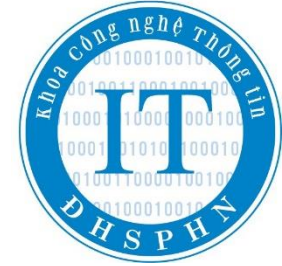
❖ **Cú pháp:**

```
for (biểu_thức_1; biểu_thức_2; biểu_thức_3) {  
    //Thực hiện công việc  
}
```

❖ **Lưu đồ:**



- **B1:** Khởi tạo biến điều kiện
- **B2:** Kiểm tra điều kiện
 - Nếu **đúng** thì thực hiện khối lệnh.
 - Nếu **sai** thì thoát khỏi vòng lặp.
- **B3:** Thay đổi biến điều kiện theo **biểu_thức_3**. Quay lại B2.



II. Cấu trúc lặp trong Java

2.1 Cấu trúc lặp for

❖ Giải thích:

- Biểu_thức_1: Khởi tạo giá trị ban đầu cho biến điều khiển của vòng lặp.
- Biểu_thức_2: Là biểu thức Logic thể hiện điều kiện để tiếp tục vòng lặp.
- Biểu_thức_3: Phép gán dùng thay đổi giá trị biến điều khiển.

❖ Nhận xét:

- Biểu_thức_1 lúc nào cũng chỉ đc tính toán 1 lần mỗi khi gọi thực hiện vòng lặp for.
- Biểu_thức_2, 3 và thân vòng for có thể thực hiện lặp lại nhiều lần.



II. Cấu trúc lặp trong Java

2.1 Cấu trúc lặp for

❖ Lưu ý:

- Nếu biểu_thức_2 không có, vòng for được xem là luôn đúng và sẽ lặp vô hạn. Muốn thoát khỏi vòng for chúng ta phải dùng 1 trong 3 lệnh **break**, **goto** hoặc **return**.
- Trong thân vòng for có thể chứa các cấu trúc điều khiển và lặp khác nhau.
- Khi gặp lệnh break, cấu trúc lặp sâu nhất sẽ thoát ra.
- Trong thân for có thể dùng lệnh **goto** để thoát khỏi vòng lặp và đến vị trí mong muốn.
- Trong thân for có thể dùng lệnh **return** để thoát khỏi vòng lặp và đến trở lại một hàm nào đó.
- Trong thân for có thể dùng lệnh **continue** để chuyển đến đầu vòng lặp và bắt đầu điều kiện lặp tiếp theo.



II. Cấu trúc lặp trong Java

2.1 Cấu trúc lặp for

- ❖ Ví dụ: Viết chương trình in ra màn hình câu “Hello world lần i” 3 lần trong đó i ($i \leq 3$) là số thứ tự mỗi lần lặp.

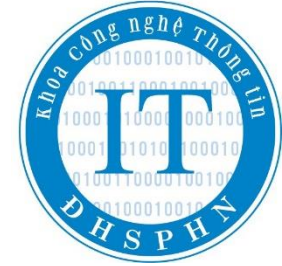
```
SinhVien.java
1 package com.hnue.myproject.package1;
2
3 import java.util.Scanner;
4
5 public class SinhVien {
6     public static void main(String[] args) {
7         for(int i = 1; i <= 3; i++)
8         {
9             System.out.println("Hello world lan " + i);
10        }
11    }
12 }
```

```
Console
<terminated> SinhVien [Java Application] C:\Users\HaiLong\.p2\pool\plugins\org.eclipse.justj.openjdk.hotspot.jre.full.win32.x86_64_14.0.2.v20200815-0932\jre\bin\javaw.exe (Oct 29, 2020, 2:24:25 PM – 2:24:30 PM)
Hello world lan 1
Hello world lan 2
Hello world lan 3
```



II. Cấu trúc lặp trong Java

2.2 Cấu trúc lặp while



II. Cấu trúc lặp trong Java

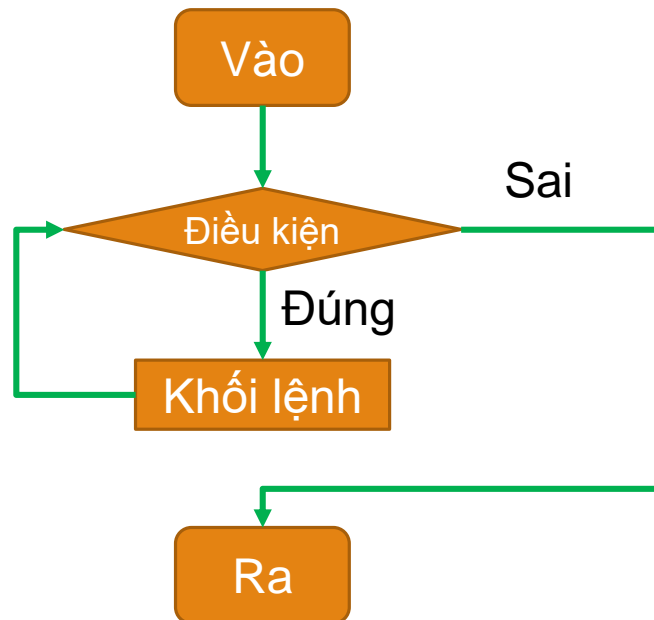
2.2 Cấu trúc lặp while

❖ Thực hiện lặp đi lặp lại công việc khi biểu thức điều kiện vẫn còn đúng.

❖ **Cú pháp:**

```
while (biểu_thức_điều_kiện) {  
    //Thực hiện công việc  
}
```

❖ **Lưu đồ:**



- Kiểm tra điều kiện
 - Nếu **đúng** thì thực hiện khối lệnh.
 - Nếu **sai** thì thoát khỏi vòng lặp.



II. Cấu trúc lặp trong Java

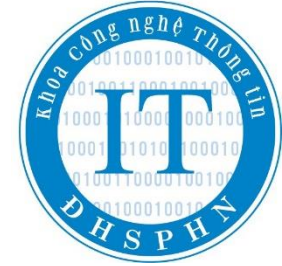
2.2 Cấu trúc lặp while

❖ Giải thích:

- biểu_thức_điều_kiện: Có thể là một biểu thức logic hoặc giá trị.

❖ Nhận xét:

- Trong thân vòng while có thể chứa các cấu trúc điều khiển khác.
- Trong thân vòng while có thể dùng lệnh continue để quay về đầu vòng lặp.
- Muốn thoát ra khỏi vòng while có thể dùng 1 trong các lệnh **break**, **goto**, **return**.



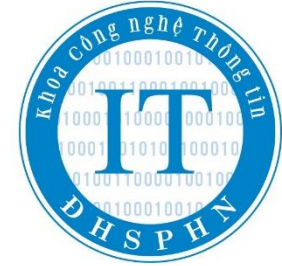
II. Cấu trúc lặp trong Java

2.2 Cấu trúc lặp while

❖ **Ví dụ:** Viết chương trình in ra màn hình câu “Hello world bằng vòng lặp while lần i” 3 lần trong đó i ($i \leq 3$) là số thứ tự mỗi lần lặp.

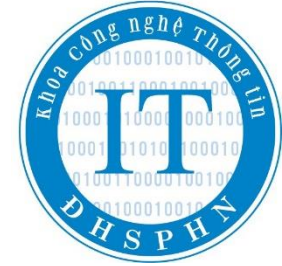
```
SinhVien.java
1 package com.hnue.myproject.package1;
2
3 public class SinhVien {
4     public static void main(String[] args) {
5         int i = 1;
6         while(i < 4) {
7             System.out.println("Hello world bang vong lap while lan " + i);
8             i++;
9         }
10    }
11 }
12
```

```
Console
<terminated> SinhVien [Java Application] C:\Users\Hailong\.p2\pool\plugins\org.eclipse.justj.openjdk.hotspot.jre.full.win32.x86_64_14.0.2.v20200815-0932\jre\bin\javaw.exe (Oct 29, 2020, 10:13:39 PM - 10:13:39 PM)
Hello world bang vong lap while lan 1
Hello world bang vong lap while lan 2
Hello world bang vong lap while lan 3
```



II. Cấu trúc lặp trong Java

2.3 Cấu trúc lặp do...while



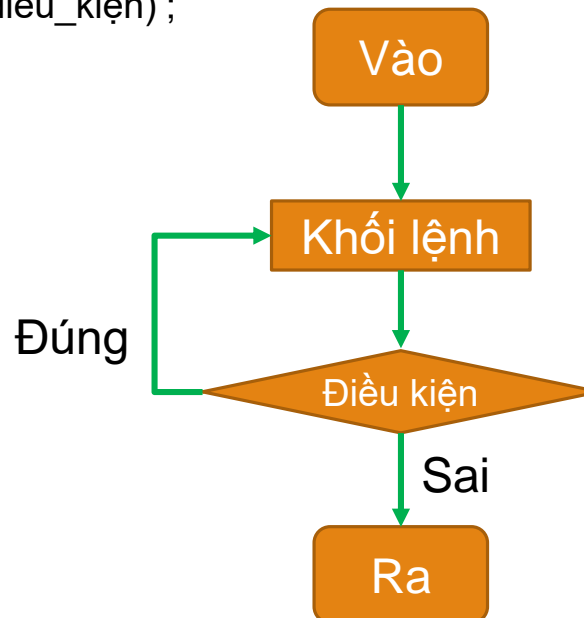
II. Cấu trúc lặp trong Java

2.3 Cấu trúc lặp do...while

❖ Thực hiện lặp đi lặp lại công việc khi biểu thức điều kiện vẫn còn đúng.

❖ **Cú pháp:** `do{
 //Thực hiện công việc
} while (biểu_thức_điều_kiện);`

❖ **Lưu đồ:**



- **B1:** Thực hiện công việc
- **B2:** Kiểm tra điều kiện
 - Nếu **đúng** thì quay lại bước B1.
 - Nếu **sai** thì thoát khỏi vòng lặp.



II. Cấu trúc lặp trong Java

2.3 Cấu trúc lặp do...while

❖ Giải thích:

- biểu_thức_điều_kiện: Có thể là một biểu thức logic hoặc giá trị.

❖ Nhận xét:

- Trong thân vòng do...while có thể chứa các cấu trúc điều khiển khác.
- Trong thân vòng do...while có thể dùng lệnh continue để quay về đầu vòng lặp.
- Muốn thoát ra khỏi vòng do...while có thể dùng 1 trong các lệnh **break**, **goto**, **return**.

❖ Lưu ý: Vòng lặp do...while sẽ thực hiện công việc ít nhất 1 lần.



II. Cấu trúc lặp trong Java

2.3 Cấu trúc lặp do...while

❖ **Ví dụ:** Viết chương trình in ra màn hình câu “Hello world bằng vòng lặp do...while lần i” 1 lần trong đó i ($i > 3$) là số thứ tự mỗi lần lặp.

```
1 package com.hnue.myproject.package1;
2
3 public class SinhVien {
4     public static void main(String[] args) {
5         int i = 1;
6         do{
7             System.out.println("Hello world bang vong lap do...while lan " + i);
8             i++;
9         }while(i > 3);
10    }
11 }
12
```

Console

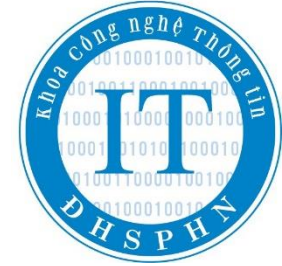
<terminated> SinhVien [Java Application] C:\Users\Hailong\.p2\pool\plugins\org.eclipse.justj.openjdk.hotspot.jre.full.win32.x86_64_14.0.2.v20200815-0932\jre\bin\javaw.exe (Oct 29, 2020, 10:43:12 PM – 10:43:12 PM)

Hello world bang vong lap do...while lan 1



II. Cấu trúc lặp trong Java

2.4 Các lệnh dừng vòng lặp



II. Cấu trúc lặp trong Java

2.4 Các lệnh dừng vòng lặp

❖ Lệnh break:

- Được sử dụng để ngắt khỏi các vòng lặp hoặc cấu trúc điều khiển khi cần thiết.
- Được sử dụng cho các bài toán không biết trước được khi nào dừng hoặc biểu thức điều kiện lặp quá phức tạp phải biểu diễn trong thân vòng lặp.



II. Cấu trúc lặp trong Java

2.4 Các lệnh dừng vòng lặp

❖ Lệnh break:

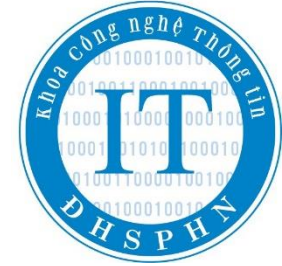
- Ví dụ: Sử dụng **break** với vòng lặp for

```
1 package com.hnue.myproject.package1;
2 public class SinhVien {
3     public static void main(String[] args) {
4         for(int i = 1; i <= 10; i++)
5         {
6             if(i > 3)
7             {
8                 break;
9             }
10            else
11            {
12                System.out.println("Hello world lan thu " + i);
13            }
14        }
15    }
16 }
```

Console

<terminated> SinhVien [Java Application] C:\Users\Hailong\p2\pool\plugins\org.eclipse.justj.openjdk.hotspot.jre.full.win32.x86_64_14.0.2.v20200815-0932\jre\bin\javaw.exe (Oct 29, 2020, 11:14:04 PM – 11:14:04 PM)

Hello world lan thu 1
Hello world lan thu 2
Hello world lan thu 3

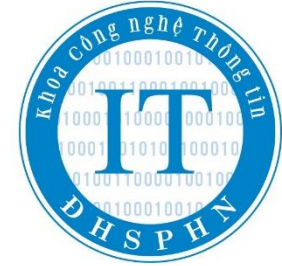


II. Cấu trúc lặp trong Java

2.4 Các lệnh dừng vòng lặp

❖ Lệnh continue:

- Được sử dụng để ngắt khỏi lần lặp hiện tại và bắt đầu lần lặp tiếp theo.
- Được sử dụng khi giá trị trong thân vòng lặp không thích hợp để thực hiện công việc nữa, cần bỏ qua để sang lần lặp tiếp theo.



II. Cấu trúc lặp trong Java

2.4 Các lệnh dừng vòng lặp

❖ Lệnh continue :

- Ví dụ: Sử dụng **continue** với vòng lặp for

```
SinhVien.java
1 package com.hnue.myproject.package1;
2 public class SinhVien {
3     public static void main(String[] args) {
4         for(int i = 1; i <= 10; i++)
5         {
6             if(i%2 == 0)
7             {
8                 continue;
9             }
10            else
11            {
12                System.out.println(i);
13            }
14        }
15    }
16 }

Console
<terminated> SinhVien [Java Application] C:\Users\Hailong\p2\pool\plugins\org.eclipse.justi.openjdk.hotspot.jre.full.win32.x86_64_14.0.2.v20200815-0932\jre\bin\javaw.exe (Oct 29, 2020, 11:17:31 PM)
1
3
5
7
9
```



THANKS FOR WATCHING