

3. Lập trình bài đầu tiên: In ra dòng chữ "Hello world" ra console, cách nhập dữ liệu từ màn hình console và in ra dòng chữ đó.

```
java                                                                    Copy

import java.util.Scanner;

public class HelloWorld {
    public static void main(String[] args) {
        // In ra dòng chữ "Hello world"
        System.out.println("Hello world");

        // Nhập dữ liệu từ màn hình console
        Scanner scanner = new Scanner(System.in);
        System.out.print("Nhập dòng chữ của bạn: ");
        String input = scanner.nextLine();

        // In ra dòng chữ vừa nhập
        System.out.println("Bạn đã nhập: " + input);
    }
}
```

4. Biến trong Java, có những loại biến nào và điểm khác nhau, quy chuẩn đặt tên biến trong Java.

- **Các loại biến trong Java:**

1. **Biến cục bộ (Local Variable):** Được khai báo bên trong một phương thức, khối lệnh hoặc hàm tạo. Chỉ có thể truy cập trong phạm vi đó.
2. **Biến instance (Instance Variable):** Được khai báo bên trong một lớp nhưng bên ngoài các phương thức. Mỗi đối tượng của lớp có một bản sao riêng của biến instance.
3. **Biến lớp (Class Variable/Static Variable):** Được khai báo với từ khóa `static`. Biến này thuộc về lớp chứ không phải đối tượng cụ thể nào.

- **Quy chuẩn đặt tên biến:**

- Tên biến nên bắt đầu bằng một chữ cái viết thường (ví dụ: `myVariable`).
- Nếu tên biến gồm nhiều từ, các từ tiếp theo nên viết hoa chữ cái đầu (ví dụ: `myVariableName`).
- Không được bắt đầu bằng số hoặc ký tự đặc biệt.
- Không được trùng với các từ khóa trong Java.



5. Kiểu dữ liệu, tìm hiểu kiểu dữ liệu nguyên thủy.

- **Kiểu dữ liệu nguyên thủy (Primitive Data Types):**

1. **byte:** Kiểu số nguyên 8-bit.
2. **short:** Kiểu số nguyên 16-bit.
3. **int:** Kiểu số nguyên 32-bit.
4. **long:** Kiểu số nguyên 64-bit.
5. **float:** Kiểu số thực 32-bit.
6. **double:** Kiểu số thực 64-bit.
7. **boolean:** Kiểu logic (true/false).
8. **char:** Kiểu ký tự 16-bit Unicode.

6. Tìm hiểu câu điều kiện (if-else và switch case) và vòng lặp trong Java (for và while, điều kiện dừng vòng lặp).

Câu lệnh if-else: Sử dụng if ls khi có hi lự chọn, một ch điều kiện đúng và một ch điều kiện sai.

```
java Copy
int x = 10;
if (x > 5) {
    System.out.println("x lớn hơn 5");
} else {
    System.out.println("x nhỏ hơn hoặc bằng 5");
}
```

Câu lệnh switch case: Câu lệnh switch case được sử dụng khi có nhiều điều kiện kiểm tra khác nhau cho một biến.

```
java                                                                    Copy

int day = 3;
switch (day) {
    case 1:
        System.out.println("Thứ Hai");
        break;
    case 2:
        System.out.println("Thứ Ba");
        break;
    // ...
    default:
        System.out.println("Không hợp lệ");
}
```

- **Vòng lặp:**

- **for:**

```
java                                                                    Copy

for (int i = 0; i < 5; i++) {
    System.out.println("i = " + i);
}
```

- **while:**

```
java                                                                    Copy

int i = 0;
while (i < 5) {
    System.out.println("i = " + i);
    i++;
}
```

- **Điều kiện dừng vòng lặp:** Vòng lặp sẽ dừng khi điều kiện kiểm tra trả về `false` .

6. Các hàm toán học trong Java.

- Java cung cấp các hàm toán học thông qua lớp `Math` :
 - `Math.sqrt(x)` : Căn bậc hai của x.
 - `Math.pow(x, y)` : x mũ y.
 - `Math.abs(x)` : Giá trị tuyệt đối của x.
 - `Math.max(x, y)` : Trả về giá trị lớn nhất giữa x và y.
 - `Math.min(x, y)` : Trả về giá trị nhỏ nhất giữa x và y.
 - `Math.random()` : Trả về một số ngẫu nhiên từ 0.0 đến 1.0.

7. Tìm hiểu hàm (function) trong Java.

- **Hàm (Function/Method):** Là một khối lệnh thực hiện một nhiệm vụ cụ thể. Hàm có thể có hoặc không có tham số và có thể trả về một giá trị hoặc không.

```
java                                                                    Copy

public class Main {
    // Hàm không có tham số và không trả về giá trị
    public static void sayHello() {
        System.out.println("Hello!");
    }

    // Hàm có tham số và trả về giá trị
    public static int add(int a, int b) {
        return a + b;
    }

    public static void main(String[] args) {
        sayHello();
        int result = add(5, 3);
        System.out.println("5 + 3 = " + result);
    }
}
```

8. Tìm hiểu về mảng (array) trong Java.

- **Mảng (Array):** Là một cấu trúc dữ liệu lưu trữ các phần tử cùng kiểu.

```
java                                                                    Copy

public class Main {
    public static void main(String[] args) {
        // Khai báo và khởi tạo mảng
        int[] numbers = {1, 2, 3, 4, 5};

        // Truy cập phần tử trong mảng
        System.out.println("Phần tử đầu tiên: " + numbers[0]);

        // Duyệt mảng
        for (int i = 0; i < numbers.length; i++) {
            System.out.println("Phần tử thứ " + i + ": " + numbers[i]);
        }
    }
}
```

9. Tìm hiểu về chuỗi (string) trong Java, cách nối 2 chuỗi.

- **Chuỗi (String):** Là một đối tượng trong Java, được sử dụng để lưu trữ và thao tác với văn bản.

```
java                                                                    Copy

public class Main {
    public static void main(String[] args) {
        // Khai báo chuỗi
        String str1 = "Hello";
        String str2 = "World";

        // Nối 2 chuỗi
        String result = str1 + " " + str2;
        System.out.println(result);

        // Sử dụng phương thức concat
        String result2 = str1.concat(" ").concat(str2);
        System.out.println(result2);
    }
}
```