



Seoul
Software
ACademy

웹 개발자 부트캠프 과정

SeSAC x CODINGOn

With. 팀 뽀빠

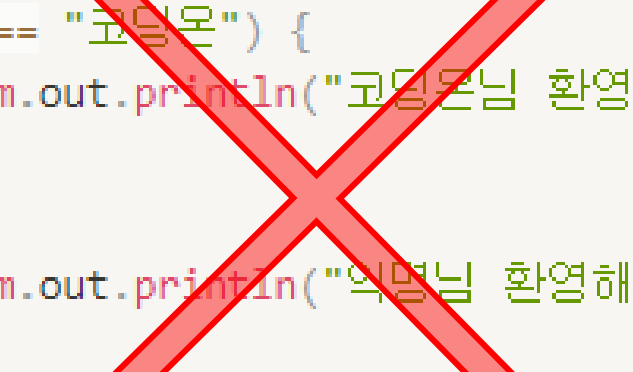
조건문

if문

```
if (number % 3 == 0) {  
    System.out.println("3의 배수입니다.");  
}  
else {  
    System.out.println("3의 배수가 아닙니다.");  
}
```

if문 - 문자열 비교

변수.equals("비교할 문자열")



```
if (name == "코딩몬") {  
    System.out.println("코딩몬님 환영해요.");  
}  
else {  
    System.out.println("익명님 환영해요.");  
}
```

```
if (name.equals("코딩몬")) {  
    System.out.println("코딩몬님 환영해요.");  
}  
else {  
    System.out.println("익명님 환영해요.");  
}
```

switch문

```
switch (number%3) {  
    case 0:  
        System.out.println("3의 배수입니다.");  
        break;  
    default:  
        System.out.println("3의 배수가 아닙니다.");  
}
```

반복문

for문

```
for(int i=0; i<10; i++) { // i가 0~9까지 10번 반복  
    System.out.print(i); // 0에서 9까지 출력  
}
```

while문

while문

```
int i = 0;
while(i<10){
    System.out.print(i);
    i++;
}
```

do-while문

```
int i = 0;
do {
    System.out.print(i);
    i++;
}
while(i<10);
```


메소드

메소드 선언

반환 값이 없는 경우

```
public class Main {  
    public static void hello() {  
        System.out.println("안녕하세요");  
    }  
  
    public static void main(String[] args) {  
        hello();  
    }  
}
```

반환 값이 있는 경우

```
public class Main {  
    public static int sum( int num1, int num2 ) {  
        return num1+num2;  
    }  
  
    public static void main(String[] args) {  
        System.out.println(sum(1,2));  
    }  
}
```

메소드 오버로딩

- 메소드 오버로딩은 한 클래스 내에서 동일한 이름의 메소드를 여러 개 정의하는 것. 이때 각 메소드는 매개변수의 타입, 개수, 순서가 달라야 함!

```
public class Calc {
    // 정수 두 개를 더하는 메소드
    public int add(int a, int b) {
        return a + b;
    }

    // 실수 두 개를 더하는 메소드
    public double add(double a, double b) {
        return a + b;
    }

    // 세 개의 정수를 더하는 메소드
    public int add(int a, int b, int c) {
        return a + b + c;
    }

    public static void main(String[] args) {
        Calc calc = new Calc();

        System.out.println(calc.add(10, 20));           // 출력: 30
        System.out.println(calc.add(10.5, 20.5));       // 출력: 31.0
        System.out.println(calc.add(10, 20, 30));       // 출력: 60
    }
}
```