

# 안드로이드 프로그래밍

2025년 2학기

# 1. 다음 Kotlin 코드의 실행 결과는 무엇인가요?

```
fun getGreeting(): String {  
    return "Hello, World!"  
}  
  
fun main() {  
    val message = getGreeting()  
    println(message)  
}
```

## 2. 다음 Kotlin 코드의 실행 결과는 무엇인가요?

```
fun add(a: Int, b: Int): Int {  
    return a + b  
}  
  
fun main() {  
    val result = add(5, 3)  
    println(result)  
}
```

### 3. 다음 Kotlin 코드의 실행 결과는 무엇인가요?

```
fun printMessage(message: String) {  
    println(message)  
}  
  
fun main() {  
    printMessage("Hello, Kotlin!")  
}
```

## 4. 다음 Kotlin 코드의 실행 결과는 무엇인가요?

```
fun calculateArea(width: Int, height: Int) = width * height

fun main() {
    val area = calculateArea(10, 5)
    println(area)
}
```

## 5. 다음 Kotlin 코드에서 main 함수 내에 적절한 함수 호출 코드를 작성하여 'Kotlin is fun!'을 출력하세요.

```
fun greet(name: String, language: String) {  
    println("$name is $language!")  
}  
  
fun main() {  
    // 여기에 코드 작성  
}
```

## 6. 다음 Kotlin 코드의 실행 결과는 무엇인가요?

```
fun getNumber(): Int {  
    return 10  
}  
  
fun main() {  
    val result = getNumber() + 5  
    println(result)  
}
```

7. 다음 Kotlin 코드에서 함수 호출 시 `message` 인자에  
`"Hello"`가, `times` 인자에 `2`가 전달되도록 함수를 호출하세요.

```
fun repeatMessage(message: String, times: Int = 1) {  
    for (i in 1..times) {  
        println(message)  
    }  
}  
  
fun main() {  
    // 여기에 코드 작성  
}
```



## 8. 다음 Kotlin 코드의 실행 결과는 무엇인가요?

```
fun sum(a: Int, b: Int): Int {  
    return a + b  
}  
  
fun main() {  
    val x = 10  
    val y = 20  
    println(sum(x, y))  
}
```

## 9. 다음 Kotlin 코드의 실행 결과는 무엇인가요?

```
fun multiply(a: Int, b: Int = 3): Int {  
    return a * b  
}  
  
fun main() {  
    val result = multiply(5)  
    println(result)  
}
```

# 함수를 이용한 간단한 계산기 구현

두 개의 정수를 입력 받아 덧셈, 뺄셈, 곱셈, 나눗셈 연산을 수행하는 함수를 구현한다. 함수를 사용하여 각 연산을 분리하고, 결과를 사용자에게 출력하시오.

1. main 함수
2. calculateSum 함수: 두 개의 정수를 인자로 받아 덧셈 결과를 반환
3. calculateDifference 함수: 두 개의 정수를 인자로 받아 뺄셈 결과를 반환
4. calculateProduct 함수: 두 개의 정수를 인자로 받아 곱셈 결과를 반환
5. calculateDivision 함수: 두 개의 정수를 인자로 받아 나눗셈 결과를 반환. 이 함수는 0으로 나누는 경우를 처리해야 한다.
6. 결과 출력: 각 연산 결과를 명확한 메시지와 함께 출력한다.

# 함수를 이용한 간단한 계산기 구현

입력과 출력 예시

입력:

- 첫 번째 숫자: 10
- 두 번째 숫자: 5

출력:

- 덧셈:  $10 + 5 = 15$
- 뺄셈:  $10 - 5 = 5$
- 곱셈:  $10 * 5 = 50$
- 나눗셈:  $10 / 5 = 2.0$

# 프로그래밍 과제 1: 문자열을 조작하는 고차 함수 만들기

목표:

고차 함수를 직접 정의하고, 람다를 인자로 전달하여 문자열을 변환하는 기능을 구현한다.

문제 설명

1. transformString이라는 이름의 고차 함수를 정의한다.
2. 이 함수는 두 개의 인자를 받는다.
  1. 첫 번째 인자: String 타입의 문자열
  2. 두 번째 인자: (String) -> String 타입의 함수 타입
3. transformString 함수는 두 번째 인자로 받은 함수를 첫 번째 인자에 적용한 결과를 반환한다.
4. main 함수에서 transformString을 호출하여 아래의 변환을 수행하고 결과를 출력하시오.
  1. 입력: "hello kotlin"
  2. 출력 1: 모든 문자를 대문자로 변환한 결과 (예: "HELLO KOTLIN")
  3. 출력 2: 공백을 밑줄(\_)로 변환한 결과 (예: "hello\_kotlin")

# 프로그래밍 과제 1: 문자열을 조작하는 고차 함수 만들기

## 입력 및 출력 예시

입력

"hello kotlin"

"hello kotlin"

적용 함수

대문자 변환 람다

공백 -> 밑줄 변환 람다

출력

"HELLO KOTLIN"

"hello\_kotlin"

# 프로그래밍 과제 2: 리스트를 필터링하는 고차 함수 활용

## 목표

코틀린 표준 라이브러리의 고차 함수 (filter, map, forEach 등)를 활용하여 컬렉션을 다루는 방법을 익힌다.

## 문제 설명

1. 정수 리스트 numbers를 선언하고 [1, 2, 3, 4, 5, 6, 7, 8, 9, 10] 값을 할당한다.
2. filter 함수와 람다를 사용하여 numbers 리스트에서 짝수만 추출하여 새로운 리스트 evenNumbers를 만든다.
3. evenNumbers 리스트의 각 요소에 map 함수를 사용하여 2를 곱한 새로운 리스트 doubledNumbers를 만든다.
4. doubledNumbers 리스트의 모든 요소를 forEach 함수와 람다를 사용하여 한 줄씩 출력한다.

# 프로그래밍 과제 2: 리스트를 필터링하는 고차 함수 활용

입력 및 출력 예시

입력	출력
[1, 2, 3, ..., 10]	4
	8
	12
	16
	20