* ${}를 사용하여 print 문 텍스트의 변수와 계산 값을 둘러쌉니다. 예를 들어 ${age}에서 age가 변수입니다.
* val 키워드와 이름을 사용하여 변수를 만듭니다. 설정이 완료되면 변경할 수 없습니다. 등호를 사용하여 변수에 값을 할당합니다. 값의 예로는 텍스트와 숫자가 있습니다.
* String은 "Hello"와 같이 따옴표로 묶인 텍스트입니다.
* Int는 양의 정수 또는 음의 정수(예: 0, 23, -1024)입니다.
* 함수에서 사용할 인수 한 개 이상을 함수에 전달할 수 있습니다. 예:fun printCakeBottom(age:Int, layers:Int) {}
* repeat() {} 문을 사용하여 일련의 명령어를 여러 번 반복합니다. 예를 들면 repeat (23) { print("%") } 또는 repeat (layers) { print("@@@@@@@@@@") }이 있습니다.
* *루프*는 명령어를 여러 번 반복하는 명령어입니다. repeat() 문은 루프의 예입니다.
* 루프를 중첩할 수 있습니다. 즉, 루프 내에 루프를 배치할 수 있습니다. 예를 들어 repeat() 문 내에 repeat() 문을 만들어 케이크 층을 만들 때처럼 여러 행에 걸쳐 기호를 여러 번 출력할 수 있습니다.

**함수 인수 사용 요약:** 함수에 인수를 사용하려면 다음 세 가지 작업을 실행해야 합니다.

* 함수 정의에 인수와 유형을 추가합니다. printBorder(border: String)
* 함수 내에서 인수를 사용합니다. println(border)
* 함수 호출 시 인수를 제공합니다. printBorder(border)

[6. 요약](https://developer.android.com/codelabs/basic-android-kotlin-training-conditional-dice-roll-behavior?continue=https%3A%2F%2Fdeveloper.android.com%2Fcourses%2Fpathways%2Fandroid-basics-kotlin-four%23codelab-https%3A%2F%2Fdeveloper.android.com%2Fcodelabs%2Fbasic-android-kotlin-training-conditional-dice-roll-behavior#5)

* if 문을 사용하여 일부 명령을 실행하는 조건을 설정합니다. 예를 들어 사용자가 행운의 숫자를 굴리면 승리 메시지를 출력합니다.
* Boolean 데이터 유형에는 true 및 false라는 값이 있고 의사 결정에 사용할 수 있습니다.
* 초과(>), 미만(<), 같음(==) 등의 연산자를 사용하여 값을 비교합니다.
* else if 문 체인을 사용하여 여러 조건을 설정합니다. 예를 들어 가능한 각 주사위 굴리기에 다른 메시지를 출력합니다.
* 조건 체인 끝에 else 문을 사용하여 명시적으로 커버되지 않을 수 있는 사례를 포착합니다. 6면 주사위 사례를 커버하면 else 문이 8면 주사위로 굴린 값인 숫자 7과 8을 포착합니다.
* 값 비교에 따라 코드를 실행하는 간단한 형식으로 when 문을 사용합니다.

**if-else의 일반 형식**

**if (*condition-is-true*) {**

***execute-this-code***

**} else if (*condition-is-true*) {**

***execute-this-code***

**} else {**

***execute-this-code***

**}**

**when 문**

**when (*variable*) {**

**matches-value -> *execute-this-code***

**matches-value -> *execute-this-code***

**...**

**}**