

01

코틀린 Overview

코틀린이란

코틀린 소개

- 스칼라(Scala, 2003), 고(Go, 2009), 다트(Dart, 2011), 스위프트(Swift, 2014) 등 다양한 언어 출시
- 코틀린은 젯브레인(<https://www.jetbrains.com/idea/>)의 오픈소스 그룹에서 개발
- 2011년에 출시, 정식 버전은 2016년에 발표
- 2017년 5월 구글 I/O 행사에서 안드로이드의 공식 언어로 코틀린을 지정
- 2025년 현재 2.1 버전

코틀린이란

Java와의 완벽한 호환성

- 기존 Java 코드와 함께 사용 가능
- Java 라이브러리 직접 사용
- 점진적 도입 가능
- 혼합 프로젝트 지원

안전한 코드

- Null 안전성 (Null Safety)
- 스마트 캐스트 (Smart Casts)
- 불변성 우선 (Immutability by default)
- 예외 처리 개선

코틀린이란

간결한 문법

- 타입 추론
- 데이터 클래스
- 확장 함수
- 람다 표현식
- 프로퍼티

코루틴 지원

- 비동기 프로그래밍 간소화
- 구조적 동시성
- Flow API

멀티플랫폼 개발

- Android/iOS 앱 개발
- 웹 프론트엔드 (Kotlin/JS)
- 서버 사이드 개발
- 네이티브 앱 (Kotlin/Native)

코틀린으로 작성 가능한 프로그램

자바를 코틀린으로

```
class User {  
    private int no;  
    private String name;  
    private String email;  
    public int getNo() {  
        return no;  
    }  
    public void setNo(int no) {  
        this.no = no;  
    }  
    public String getName() {  
        return name;  
    }  
    public void setName(String name) {  
        this.name = name;  
    }  
    public String getEmail() {  
        return email;  
    }  
    public void setEmail(String email) {  
        this.email = email;  
    }  
}
```

```
data class User(var no: Int, var name: String, var email: String)
```

코틀린으로 작성 가능한 프로그램

안드로이드

```
public class MainActivity extends AppCompatActivity implements View.OnClickListener{
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);

        Button testBtn=(Button)findViewById(R.id.testBtn);
        testBtn.setOnClickListener(this);
    }
    @Override
    public void onClick(View view) {
        Toast.makeText(this, "button click!!", Toast.LENGTH_SHORT).show();
    }
}
```

```
class MainActivity : AppCompatActivity(), View.OnClickListener{
    override fun onCreate(savedInstanceState: Bundle?) {
        super.onCreate(savedInstanceState)
        setContentView(R.layout.activity_main)

        testBtn.setOnClickListener(this)
    }
    override fun onClick(p0: View?) {
        Toast.makeText(this, "Button Click!!", Toast.LENGTH_SHORT).show()
    }
}
```

코틀린으로 작성 가능한 프로그램

자바스크립트

```
window.onload=function(){  
    var btn=document.getElementById("btn");  
    var result=document.getElementById("result");  
    btn.addEventListener('click', function(){  
        result.innerText='Hello Kkang';  
    });  
}
```

```
fun onLoad(){  
    val btn = document.getElementById("btn")!! as HTMLElement  
    val result=document.getElementById("result")!! as HTMLElement  
    btn.onclick= {  
        result.appendText("Hello Kkang")  
    }  
}  
  
fun main(args: Array<String>) {  
    window.onload = {  
        onLoad()  
    }  
}
```

코틀린으로 작성 가능한 프로그램

백엔드

```
@RestController
public class GreetingController {

    AtomicLong counter=new AtomicLong();

    @GetMapping("/greeting")
    public Greeting greeting(@RequestParam(value="name", defaultValue = "World") String name){
        return new Greeting(counter.incrementAndGet(), "Java, "+name);
    }
}
```

```
@RestController
class GreetingController {

    val counter = AtomicLong()

    @GetMapping("/greeting")
    fun greeting(@RequestParam(value = "name", defaultValue = "World") name: String) =
        Greeting(counter.incrementAndGet(), "Hello, $name")
}
```


코틀린 주요 프레임워크

Spring Boot

- 자바 기반의 백엔드 애플리케이션 개발 프레임워크
- 2017/12, 스프링 공식 언어로 지정

Ktor

- JetBrains 에서 만든 코틀린 기반의 웹 애플리케이션 개발 프레임워크

Kotlin Native

- 크로스 플랫폼 애플리케이션 개발 기술, iOS, macOS, windows, Linux 등

KMM (Kotlin Multiplatform Mobile)

- Kotlin Native 에서 사용하는 개발 프레임워크
- iOS 앱 개발에 사용이 목적

개발환경

IDE 지원

- IntelliJ IDEA (Ultimate/Community)
- Android Studio
- Eclipse (플러그인)
- Visual Studio Code (플러그인)

빌드 도구

- Gradle
- Maven
- Ant



감사합니다

단단히 마음먹고 떠난 사람은
산꼭대기에 도착할 수 있다.
산은 올라가는 사람에게만 정복된다.



윌리엄 셰익스피어
William Shakespeare