

01

01-9. 안드로이드에서 코루틴 활용

Coroutine

## 메인 스레드와 코루틴 디스패처

---

- 액티비티, 프래그먼트는 화면을 목적
- 액티비티, 프래그먼트는 메인 스레드에 의해 실행되며 화면 출력이 되고 유저 이벤트를 처리
- 코루틴에 의해 비동기 업무가 진행되더라도 화면 관련된 작업은 Main 스레드에 의해 처리되게 작성

## 라이프사이클 제어

---

- 안드로이드는 하나의 애플리케이션 내에 여러 액티비티, 서비스등의 컴포넌트를 가지며 각각의 컴포넌트들은 생성부터 소멸까지의 라이프사이클을 가진다.
- 코루틴을 구동시킨 곳이 종료될 때 코루틴이 같이 종료되지 않을 가능성 고려

## 액티비티에서 lifecycleScope 이용

---

- 액티비티가 종료될 때 같이 종료되는 코루틴을 위한 lifecycleScope 라는 스코프가 제공

```
dependencies {  
    implementation 'androidx.lifecycle:lifecycle-runtime-ktx:2.6.1'  
}
```

## 프래그먼트에서 `lifecycleScope` 이용

---

- 프래그먼트에서는 `lifecycleScope` 와 `viewLifecycleOwner.lifecycleScope` 두개를 제공
- `lifecycleScope` 는 프래그먼트 자체의 생명주기와 연결되어 있어서 프래그먼트가 종료(`onDestory()` 함수 호출 되는 시점)되어야 코루틴도 취소
- `viewLifecycleOwner.lifecycleScope` 는 `viewLifecycleOwner` 가 프래그먼트 자체가 아니라 프래그먼트에서 출력하는 뷰의 생명주기를 가지고 있기 때문에 `onDestoryView()` 함수가 호출되게 되면 `viewLifecycleOwner.lifecycleScope` 로 실행시킨 코루틴도 같이 취소

# viewModelScope

---

- 뷰모델 생명주기와 연결된 스코프에서 코루틴을 구동을 위해 viewModelScope 를 제공

```
dependencies {  
    implementation 'androidx.lifecycle:lifecycle-viewmodel-ktx:2.6.1'  
}
```

```
viewModelScope.launch {  
  
}
```

## liveData { }, LiveDataScope

---

- LiveData 의 생존주기를 이용하는 코루틴이 필요
- LiveData 의 데이터를 구독하는 곳과 관련이 없이 코루틴이 실행되고 데이터를 발행할 가능성
- LiveData 의 생존주기를 이용하는 코루틴을 위해 LiveDataScope 가 제공되며 LiveDataScope 에서 실행되는 코루틴을 만들기 위해서 liveData() 가 제공

```
dependencies {  
    implementation 'androidx.lifecycle:lifecycle-livedata-ktx:2.6.1'  
}
```



# 감사합니다

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



윌리엄 셰익스피어  
William Shakespeare