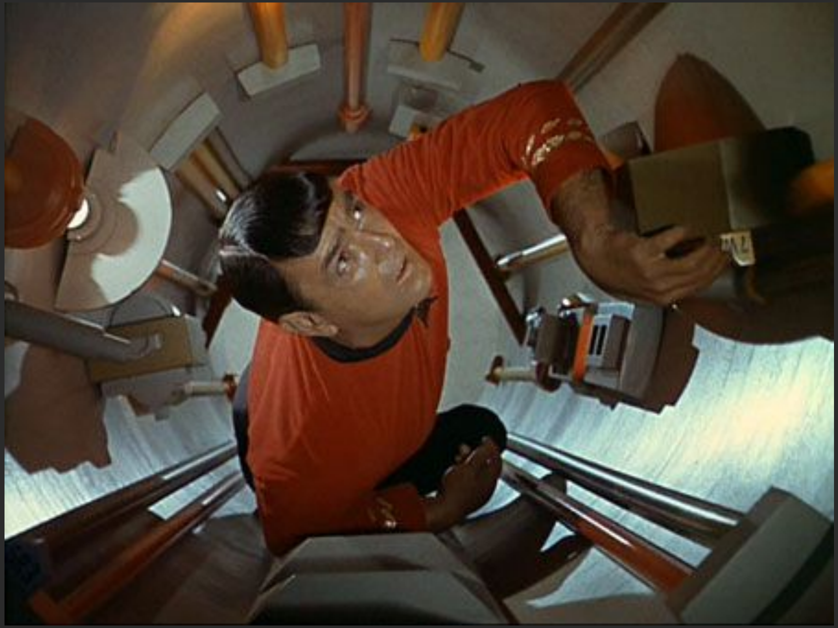
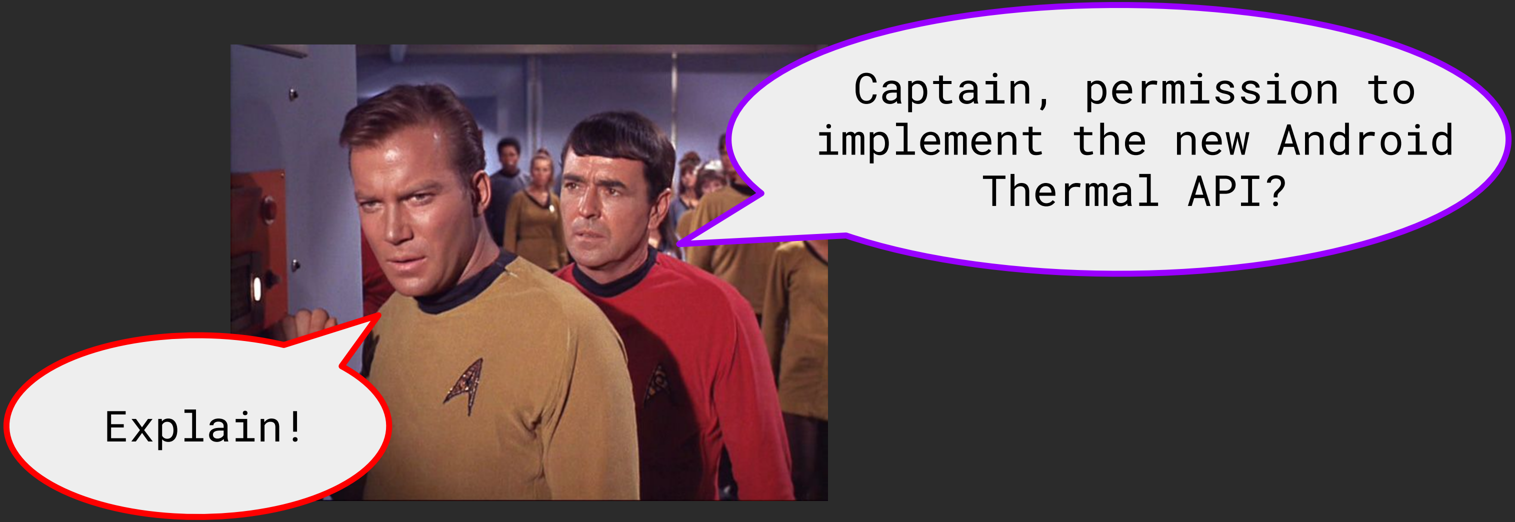


Thermal State

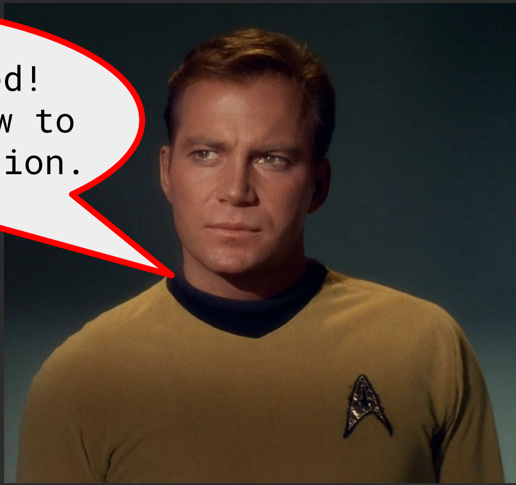
By Moro

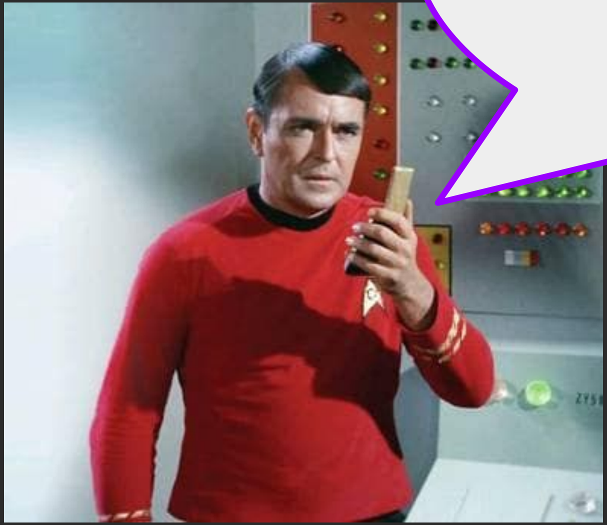

Captain, permission to implement the new Android Thermal API?
Explain!

Android Q introduces a **new thermal API**. We have some apps making intensive processing, recording videos, and other heavy tasks. To avoid unexpected results, this API could help us.

Permission granted! Keep logs about how to do this implementation.

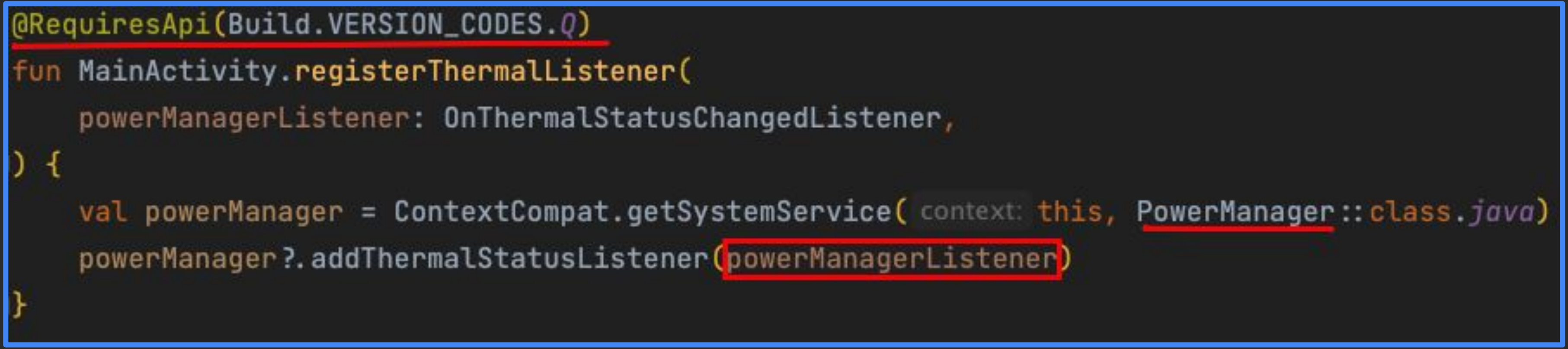
Aye Sir!





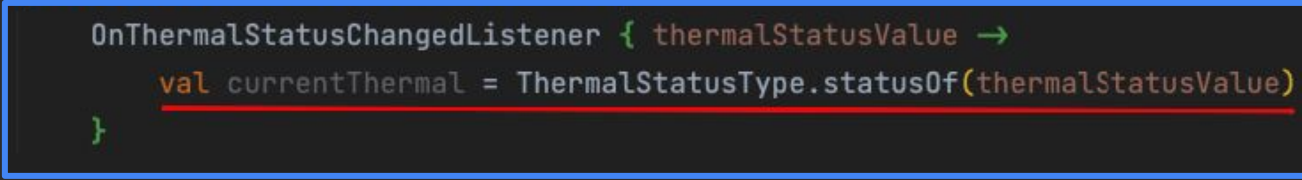
Log entry. Scotty, star date 2050.
I will record step by step about how to implement the new Thermal Status API on Android.

First, register the thermal status listener. A good place to add is in our *onCreate* method.

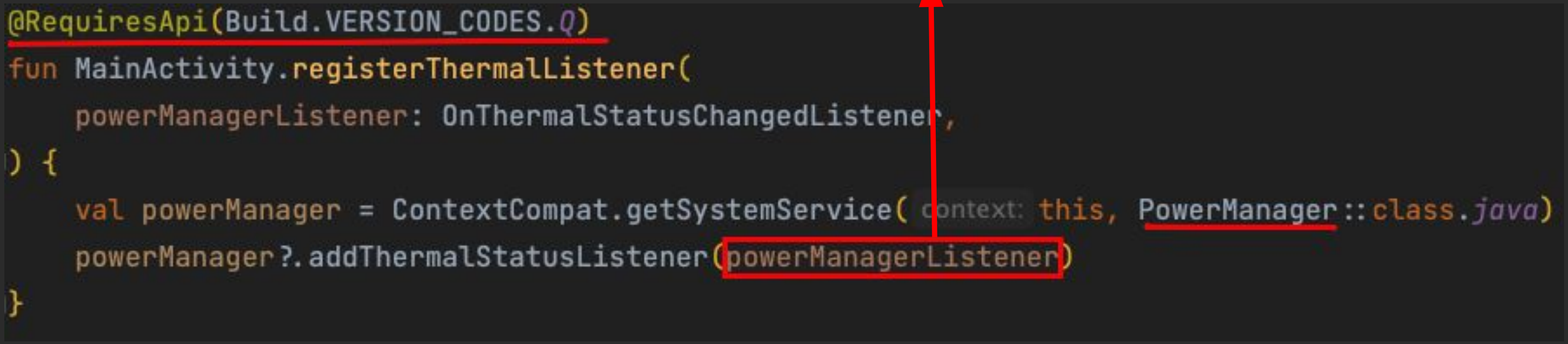


```kotlin
@RequiresApi(Build.VERSION_CODES.Q)
fun MainActivity.registerThermalListener(
    powerManagerListener: OnThermalStatusChangedListener,
) {
    val powerManager = ContextCompat.getSystemService( context: this, PowerManager::class.java)
    powerManager?.addThermalStatusListener(powerManagerListener)
}
```

Second, specify the listener. This listener will be called every time the thermal status change.

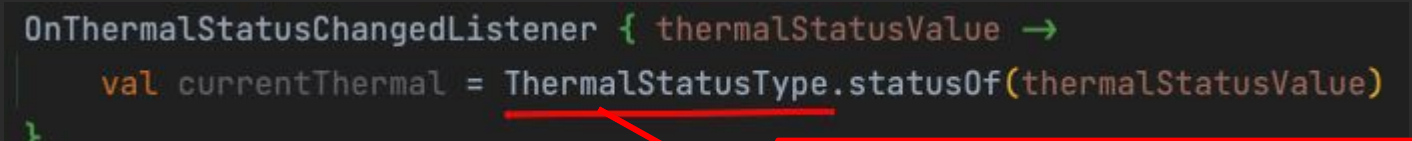


```
OnThermalStatusChangedListener { thermalStatusValue ->
    val currentThermal = ThermalStatusType.statusOf(thermalStatusValue)
}
```

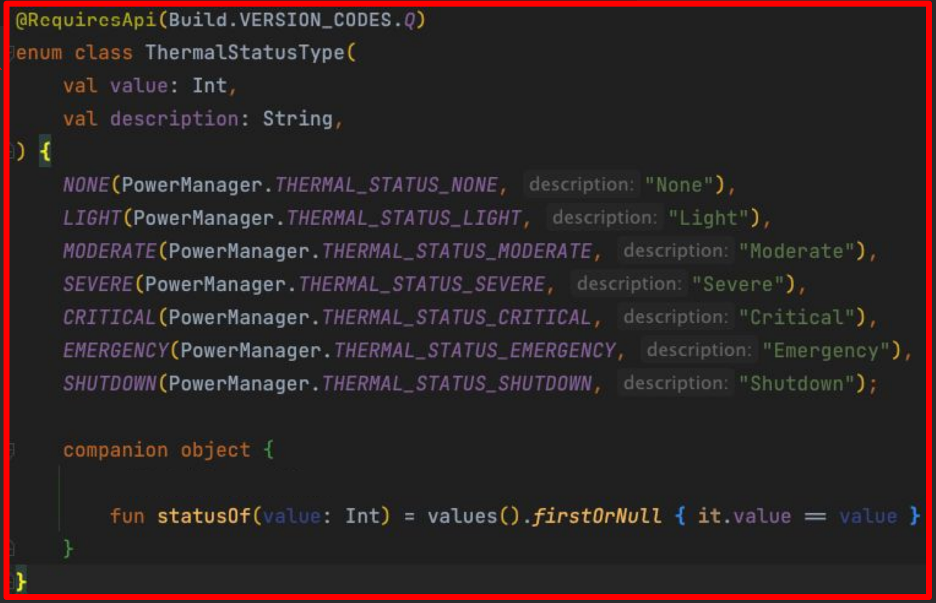


```
@RequiresApi(Build.VERSION_CODES.Q)
fun MainActivity.registerThermalListener(
    powerManagerListener: OnThermalStatusChangedListener,
) {
    val powerManager = ContextCompat.getSystemService( context: this, PowerManager::class.java)
    powerManager?.addThermalStatusListener(powerManagerListener)
}
```

You can create an enum type to represent each Thermal Status.



```kotlin
OnThermalStatusChangedListener { thermalStatusValue ->
    val currentThermal = ThermalStatusType.statusOf(thermalStatusValue)
}
```



```kotlin
@RequiresApi(Build.VERSION_CODES.Q)
enum class ThermalStatusType(
    val value: Int,
    val description: String,
) {
    NONE(PowerManager.THERMAL_STATUS_NONE, description: "None"),
    LIGHT(PowerManager.THERMAL_STATUS_LIGHT, description: "Light"),
    MODERATE(PowerManager.THERMAL_STATUS_MODERATE, description: "Moderate"),
    SEVERE(PowerManager.THERMAL_STATUS_SEVERE, description: "Severe"),
    CRITICAL(PowerManager.THERMAL_STATUS_CRITICAL, description: "Critical"),
    EMERGENCY(PowerManager.THERMAL_STATUS_EMERGENCY, description: "Emergency"),
    SHUTDOWN(PowerManager.THERMAL_STATUS_SHUTDOWN, description: "Shutdown");

    companion object {

        fun statusOf(value: Int) = values().firstOrNull { it.value == value }
    }
}
```

Status	Effect	
None	Not under throttling	🤗
Light	Light throttling where UX is not impacted	😅
Moderate	Moderate throttling where UX is not largely impacted	🙃
Severe	Severe throttling where UX is largely impacted	☹️
Critical	Platform has done everything to reduce power	🤨
Emergency	Key components in platform are shutting down due to thermal condition.	😨
Shutdown	Need shutdown immediately	😱

By Moro