

로컬 알림

- 알림센터를 통해 전달하는 방법
- iOS 스케줄러에 의해 발송
- 모든 발송의 스케줄러에 의해 관리

UserNotification 가 만들어지면서

UILocalNotification 객체는 iOS 10까지만 사용

```
func application(_ application: UIApplication, didFinishLaunchingWithOptions launchOptions: [UIApplicationLaunchOptionsKey: Any]?) -> Bool
```

는 앱이 처음 실행될때 호출되는 메소드

상세 시점

- 애플리케이션에서 사용할 클래스와 리소스들이 모두 메모리에 로드되고 아직 애플리케이션의 첫 화면을 모바일 디바이스에 띄우기 직전, 론치 스크린이 표시되고있는동안 호출

로컬알림또는 푸시알림을 사용하기 위해서는 UIUserNotificationSettings 객체를 이용하여 설정환경 정의 이를 등록해야한다.

registerUserNotificationSettings() 메소드를 통해

설정내용 바탕으로 사용자 동의창 생성 하고 동의를 통해 알림 환경 등록

applicationWillResignActive(_:)

앱이 활성화 상태를 잃었을때 실행되는 메소드

(예시. 사용중 전화, SMS 도착하는 경우, 홈버튼을 눌러 앱을 백그라운드로 내렸을 때가 대표적인 경우 이때 호출됨)

if #available() 구문: 플랫폼 버전에 따라 실행 로직 나누는 구문

currentUserNotificationSettings 속성을 통해 현재 설정되어있는 알림 설정값을 읽어들이

UIApplication.shared(앱 델리게이트에서는 application 매개변수)

`.registerUserNotificationSettings(_:)`

알림 허용여부를 사용자에게 확인받고, 사용자의 선택을 애플리케이션에 등록하는 역할을 합니다.

`.currentUserNotificationSettings`

현재 설정된 알림 허용 여부 정보를 읽어옵니다.

`.scheduleLocalNotification(_:)`

생성된 알림 객체를 iOS의 스케줄러에 등록합니다. 등록된 알림 객체는 `fireDate` 속성에 설정된 시간에 맞게 발송됩니다.

`.presentLocalNotificationNow(_:)`

생성된 알림 객체의 `fireDate` 속성을 무시하고, 즉각 발송합니다.

앱이 실행중인 상태에서 도착한다면 활성화/비활성화 상태와 관계없이 앱 델리게이트의 `application(_:didReceive)`메소드가 호출된다.
이때 알림 객체 정보도 함께 전달

UserNotifications 프레임워크

UNMutableNotificationContent 는 알림에 필요한 메시지와 같은 기본적인 속성을 담은 콘텐츠 역할

로컬 알림 타이틀, 서브 타이틀, 알림 메시지 설정

UNNotificationContent 위 객체와 비슷, 하지만, 이객체는 수정이 불가능 read-only

UNTimeIntervalNotificationTrigger는 알림 발송조건 관리

만약 하루중 특정시각에 맞추려면

UNCalendarNotificationTrigger객체를 사용해야한다.

```
// UNNotificationRequest 클래스를 초기화 하여 알림 요청 객체를생성. 알림요청 객체의 아이디로 설정된 "wakeup"은 등록된 알림 요청 취소 시 사용
let request = UNNotificationRequest(identifier: "WakeUp", content: nContent, trigger: calendarTrigger)

// 생성된 알림 요청 객체를 노티피케이션 센터에 추가. 노티피케이션 센터는 iOS의 스케줄링 센터에 이값을 등록, 정해진 시간에 발송되도록 처리
UNUserNotificationCenter.current().add(request, withCompletionHandler: { (_) in
    let message = "알림이 등록되었습니다., 등록된 알림은 매일 \(hourKeyString!):\(\minuteKeyString!) 마다 알림이 옵니다."

    let alert = UIAlertController(title: "알림 등록", message: message, preferredStyle: .alert)

    let ok = UIAlertAction(title: "확인", style: .default, handler: { (_) in
        self.settingMainTableView.reloadData()
    })

    alert.addAction(ok)
    self.present(alert, animated: false)
})
```