

---

# Image Picker

---

강사 주영민

# UIImagePickerController

---

- 다양한 소스로부터 사진을 가져오는 기능이 구현되어있는 클래스
- 사진 소스에는 라이브러리, 사진앨범, 카메라 등이 있다.
- 옵션을 주어 사진을 가져올 때 사용자에게 정사각형으로 편집할 수 있도록 할 수 있다.

# 사용자로부터 접근 권한 요청하기

---

- ios 10에서 부터 사용자의 데이터에 접근하기 위해서는 접근 권한을 설정해야된다. (info.plist에 사용 목적 표시)
- 항목
  - ✓ 미디어 라이브러리 접근 : Privacy - Music Usage Description
  - ✓ 블루투스 인터페이스 접근 : Privacy - Bluetooth Peripheral Usage Description
  - ✓ 달력 접근 : Privacy - Calendars Usage Description
  - ✓ 카메라 접근 : Privacy - Camera Usage Description
  - ✓ 위치정보 접근 (사용할 경우만) : Privacy - Location Always and When In...
  - ✓ 사진 라이브러리 접근 : Privacy - Photo Library Additions Usage Description
  - ✓ ...등등

# 사용 순서

---

- UIImagePickerController 인스턴스 생성
- Delegate 설정  
(UIImagePickerControllerDelegate, UINavigationControllerDelegate 두개의!! 프로토콜 채택)
- UIImagePickerController의 sourceType 설정
- Present!!

# Sample Code

---

```
func showImagePicker(_ sender: Any) {  
    //1  
    let cameraController = UIImagePickerController()  
    //2 UIImagePickerControllerDelegate  
    // UINavigationControllerDelegate 두개 모두 채택!  
    cameraController.delegate = self  
    //3 sourceType설정  
    cameraController.sourceType = .photoLibrary  
    //present  
    present(cameraController, animated: true, completion: nil)  
}
```

```
enum UIImagePickerControllerSourceType : Int {  
  
    case photoLibrary  
    case camera  
    case savedPhotosAlbum  
}
```

# Sample Code - delegate

---

//사진을 선택 후 불리는 델리게이트 메소드

```
func imagePickerController(_ picker: UIImagePickerController,
didFinishPickingMediaWithInfo info: [String : Any]) {

    print(info)

    picker.dismiss(animated: true, completion: nil)

}
```

//취소했을때 불리는 델리게이트 메소드

```
func imagePickerControllerDidCancel(_ picker:
UIImagePickerController) {
    picker.dismiss(animated: true, completion: nil)
}
```

\*dismiss를 해야 닫을 수 있다.

# info - print

---

```
print(info)
```

```
["UIImagePickerControllerImageURL": file:///Users/youngminjoo/Library/Developer/
CoreSimulator/Devices/DB57D4A3-0556-4A1C-8B73-087E51A2567E/data/Containers/Data/
Application/C82402D5-CAD4-49DB-9EB2-DCB7326D9BAD/tmp/A041D3B5-38ED-4535-A80B-
B07539F2D7B7.jpeg,

"UIImagePickerControllerMediaType": public.image,

"UIImagePickerControllerReferenceURL": assets-library://asset/asset.JPG?id=9F983DBA-
EC35-42B8-8773-B597CF782EDD&ext=JPG,

"UIImagePickerControllerOriginalImage": <UIImage: 0x6040002a65a0> size {3000, 2002}
orientation 0 scale 1.000000]
```

# Info – key

---

let UIImagePickerControllerMediaType: String

Specifies the media type selected by the user.

let UIImagePickerControllerImageURL: String

A key containing the URL of the image file.

let UIImagePickerControllerEditedImage: String

Specifies an image edited by the user.

let UIImagePickerControllerCropRect: String

Specifies the cropping rectangle that was applied to the original image.

let UIImagePickerControllerMediaURL: String

Specifies the filesystem URL for the movie.

let UIImagePickerControllerMediaMetadata: String

Metadata for a newly-captured photograph.

let UIImagePickerControllerLivePhoto: String

The Live Photo representation of the selected or captured photo.

let UIImagePickerControllerPHAsset: String

The key to use when retrieving a Photos asset for the image.



# Sample Code

---

```
func imagePickerController(_ picker: UIImagePickerController,
didFinishPickingMediaWithInfo info: [String : Any]) {
    print(info)

    if let image = info[UIImagePickerControllerOriginalImage]
as? UIImage
    {
        self.displayImageView.image = image
    }
    picker.dismiss(animated: true, completion: nil)
}
```