In-class Activity on WKWebView with ActivityIndicatorView:

1. Create a single view app with XCode and name it as **myWKWebViewApp**.

   Add a **WKWeb View** into the View Controller and resize it to cover View area. Next add an **Activity Indicator View** and place it in the middle *on top of the WKWeb View's coverage area*.

   OR

   Download **myWKWebViewApp** directly from our In-class activity folder on Moodle. Open it with Xcode and use it for the rest of this in-class activity.

2. Connect the WKWeb View and Activity Indicator View objects from the View Controller to its associated file (ViewController.swift) as new referencing outlets using a name like **siteView** and **loadSpinner** respectively.

```
import UIKit

class ViewController: UIViewController {
    @IBOutlet var siteView: WKWebView!
    @IBOutlet weak var loadSpinner: UIActivityIndicatorView!

    override func viewDidLoad() {
        super.viewDidLoad()
        // Do any additional setup after loading the view.
    }
}
```

3. Let's first try to load a web site (e.g. google.com) without using an ActivityIndicator:

```
import UIKit

class ViewController: UIViewController {
    @IBOutlet var siteView: WKWebView!
    @IBOutlet weak var loadSpinner: UIActivityIndicatorView!

    override func viewDidLoad() {
        super.viewDidLoad()
        // Do any additional setup after loading the view.
        loadSpinner.isHidden = true
        loadAddress()

        siteView = WKWebView()

    }
```

```
    func loadAddress() {
        let myURL = URL(string:"https://www.google.com")
        let request = URLRequest(url: myURL!)
        siteView.load(request)
    }
}
```

4. Run the app. You might find that loading a web page may be slow. So having an activity indicator would be nice because it can show the user that the web page is still loading!

5. Let's try to use the activity indicator for the app. We need to show the activity indicator with animation effect at different stages of the loading process from the WKWeb View object. To achieve that we need to implement two protocols named **WKNavigationDelegate** and **WKUIDelegate** which provide a few event handlers for us to work with the WKWeb view. We can start and stop the animation effect of the Activity Indicator using a couple event handlers while the Web View object is loading the web page.

```
import UIKit

class ViewController: UIViewController, WKNavigationDelegate, WKUIDelegate  {
    @IBOutlet var siteView: UIWebView!
    @IBOutlet weak var loadSpinner: UIActivityIndicatorView!

    override func viewDidLoad() {
        super.viewDidLoad()
        // Do any additional setup after loading the view.
        loadSpinner.isHidden = false
        loadAddress()

        siteView.navigationDelegate = self
        self.view.addSubview(loadSpinner)
        siteView = WKWebView()

    }

    func loadAddress() {
        let myURL = URL(string:"https://www.google.com")
        let request = URLRequest(url: myURL!)
        webView.load(request)
    }

    func webView (_ webView: WKWebView, didStartProvisionalNavigation  navigation:
WKNavigation! ) {
        print("Start to Load")
        loadSpinner.startAnimating()
    }

    func webView (_ webView: WKWebView, didFinish  navigation: WKNavigation! ) {
```

```
        print("Finish to Load")
        loadSpinner.startAnimating()
        loadSpinner.isHidden = true
    }
}
```

6. Run the app! Is the Activity Indicator too small? Okay, we can do the following to make it bigger (3, 4.. times bigger) !

```
import UIKit

class ViewController: UIViewController, WKNavigationDelegate, WKUIDelegate  {
    @IBOutlet var siteView: UIWebView!
    @IBOutlet weak var loadSpinner: UIActivityIndicatorView!

    override func viewDidLoad() {
        super.viewDidLoad()
        // Do any additional setup after loading the view.
        loadSpinner.isHidden = false
        loadSpinner.scaleIndicator(factor: 3)
        loadAddress()

        siteView.navigationDelegate = self
        self.view.addSubview(loadSpinner)
        siteView = WKWebView()

    }

    func loadAddress() {
        let myURL = URL(string:"https://www.google.com")
        let request = URLRequest(url: myURL!)
        webView.load(request)
    }

    func webView (_ webView: WKWebView, didStartProvisionalNavigation  navigation:
WKNavigation! ) {
        print("Start to Load")
        loadSpinner.startAnimating()
    }

    func webView (_ webView: WKWebView, didFinish  navigation: WKNavigation! ) {
        print("Finish to Load")
        loadSpinner.startAnimating()
        loadSpinner.isHidden = true
    }
}
extension UIActivityIndicatorView {
    func scaleIndicator(factor: CGFloat) {
```

```
        transform = CGAffineTransform(scaleX: factor, y: factor)
    }
}
```

7. Run the app again! Take a screen with the bigger activity indictor running and submit it.