

IODClient Library  
for iOS – SWIFT  
Version 1.0

## Contents

1. Overview .....	3
2. Integrate IODClient into iOS Swift project .....	3
3. IODClient API reference .....	3
3.1. IODClient(apiKey:String, version:String = "v1"); .....	3
3.2. GetRequest(inout params:Dictionary<String, AnyObject>, iodApp:String, requestMode:REQ_MODE = .ASYNC) .....	4
3.3. PostRequest(inout params:Dictionary<String, Object>, iodApp:String, requestMode:REQ_MODE = .ASYNC) .....	5
3.4. GetJobResult(String jobId) .....	6

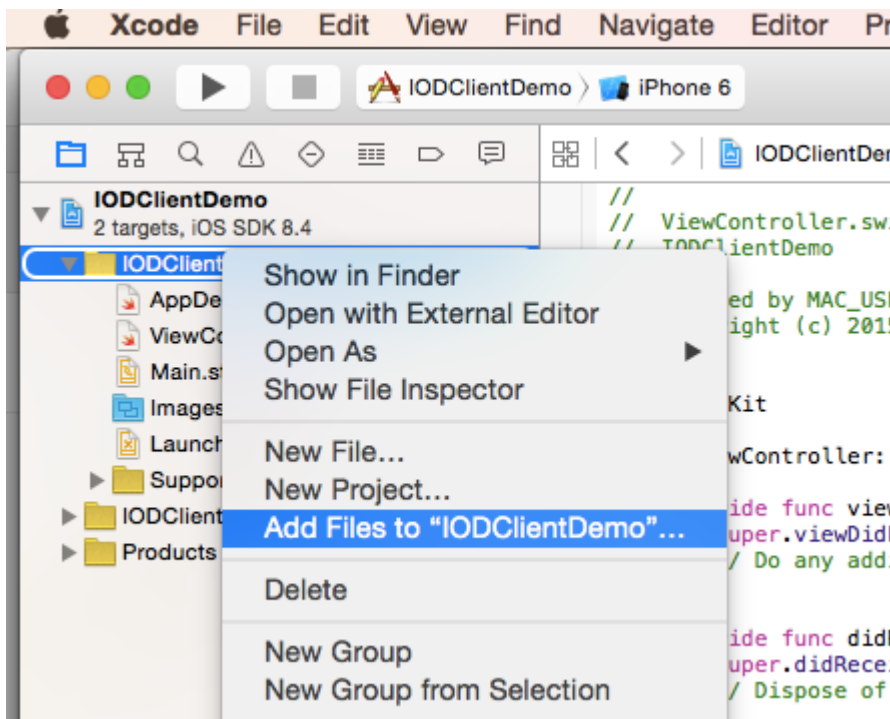
## 1. Overview

IODClient for iOS - Swift is a utility class, which helps you easily integrate your iOS app with HP IDOL OnDemand Services.

IODClient class exposes source code so you can modify it as you wish.

## 2. Integrate IODClient into iOS Swift project

- a) Download the IODClient library for iOS.
- b) Create a new or open an existing iOS Swift project
- c) Add the IODClient.swift file to the project.



- d) Browse to the folder where you saved the library and select the IODClient.swift file.

## 3. IODClient API reference

### 3.1. `IODClient(apiKey:String, version:String = "v1");`

#### Description:

Constructor. Creates and initializes an IODClient object

#### Parameters:

**apiKey:** your developer apikey.

**version:** IDOL OnDemand API version. Currently it only supports version 1. Thus, the default value is "v1".

#### Example code:

```
class MyViewController : UIViewController, IODClientDelegate {  
    var iodClient:IODClient = IODClient(apiKey: "your-api-key");  
  
    // implement delegated functions  
    func iodClient_requestCompletedWithJobID(response:String){ }  
    func iodClient_requestCompletedWithContent(response:String){ }  
    func iodClient_onErrorOccurred(errorMessage:String){ }  
}
```

### 3.2. [GetRequest\(inout params:Dictionary<String, AnyObject>, iodApp:String, requestMode:REQ\\_MODE = .ASYNC\)](#)

#### Description:

Sends a GET request to an IDOL OnDemand API.

#### Parameters:

**params:** a Dictionary object containing key/value pair parameters to be sent to an IDOL OnDemand API, where the keys are the parameters of an IDOL OnDemand API.

#### Note:

In the case of a parameter type is an array<>, the key must be defined as "arrays" and the value must be a Dictionary<String,String> object with the key is the parameter name and the values separated by commas ",". E.g.:

```
var arrays = Dictionary<String, String>()  
arrays["entity_type"] = "people_eng,places_eng"  
params["arrays"] = arrays
```

**iodApp:** a string to identify an IDOL OnDemand API. E.g. "extractentities". Current supported apps are listed in the IODClient.iodApps object.

**mode** [REQ\_MODE.SYNC | REQ\_MODE.ASYNC]: specifies API call as Asynchronous or Synchronous.

Return: void.

#### Example code:

```

var iodApp = iodClient.iodApps.ENTITY_EXTRACTION;
var arrays = Dictionary<String, String>()
arrays["entity_type"] = "people_eng,places_eng"
var params = Dictionary<String, AnyObject>()
params["url"] = http://www.cnn.com
params["arrays"] = arrays

iodClient.GetRequest(&params, iodApp:iodApp, requestMode:
IODClient.REQ_MODE.SYNC);

```

#### Response:

- If the mode is "ASYNC", response will be returned via the requestCompletedWithJobID(String response) callback function.
- If the mode is "SYNC", response will be returned via the requestCompletedWithContent(String response) callback function.
- If there is an error occurred, the error message will be sent via the onErrorOccurred(String errorMessage) callback function.

### 3.3. [PostRequest\(inout params:Dictionary<String, Object>, iodApp:String, requestMode:REQ\\_MODE = .ASYNC\)](#)

#### Description:

Sends a POST request to an IDOL OnDemand API.

#### Parameters:

**params:** a Dictionary object containing key/value pair parameters to be sent to an IDOL OnDemand API, where the keys are the parameters of an IDOL OnDemand API.

#### Note:

In the case of a parameter type is an array<>, the key must be defined as "arrays" and the value must be a Dictionary<String,String> object with the key is the parameter name and the values separated by commas ",". E.g.:

```

var arrays = Dictionary<String, String>()
arrays["entity_type"] = "people_eng,places_eng"
params["arrays"] = arrays

```

**iodApp:** a string to identify an IDOL OnDemand API. E.g. "ocrdocument". Current supported apps are listed in the IODClient.iodApps object.

**mode** [REQ\_MODE.SYNC | REQ\_MODE.ASYNC]: specifies API call as Asynchronous or Synchronous.

Return: void.

Example code:

```
var iodApp = iodClient.iodApps.OCR_DOCUMENT;
var params = Dictionary<String, Object>()
params["file"] = "full/path/filename.jpg"
params["mode"] = "document_photo"

iodClient.PostRequest(&params, iodApp:iodApp, requestMode:
IODClient.REQ_MODE.ASYNC);
```

Response:

- If the mode is "ASYNC", response will be returned via the requestCompletedWithJobID(String response) callback function.
- If the mode is "SYNC", response will be returned via the requestCompletedWithContent(String response) callback function.
- If there is an error occurred, the error message will be sent via the onErrorOccurred(String errorMessage) callback function.

### 3.4. [GetJobResult\(String jobId\)](#)

Description:

Sends a request to IDOL OnDemand to retrieve the content identified by the jobId.

Parameter:

**jobID:** the job ID returned from an IDOL OnDemand API upon an asynchronous call.

Return: void.

Response:

Response will be returned via the requestCompletedWithContent(String response)

Example code:

```
func requestCompletedWithJobID(response:String)
{
    var resStr =
response.stringByTrimmingCharactersInSet(NSCharacterSet.whitespaceChar
acterSet())

    var jsonError: NSError?

    let data = (resStr as
NSString).dataUsingEncoding(NSUTF8StringEncoding);
```

```
        let json = NSJSONSerialization.JSONObjectWithData(data!,  
options: nil, error: &jsonError) as! NSDictionary
```

```
        if let unwrappedError = jsonError {  
            println("json error: \(unwrappedError)")  
        } else {  
            var jobId = json.valueForKey("jobID") as! String;  
            iodClient.GetJobResult(jobId);  
        }  
    }
```

```
}
```