

2nd Authenticator App to App Usage Guide

- ※ Load the ExcelKEY (**excelkeylib_v1.0.aar**) library.
- ※ **API_2ND_AUTH_KEY** : '2nd Authentication API Key' must be issued from 'BaasID'.
- ※ **AccessToken** : Token issued by eXcelKEY upon Register 2nd Authentication.

I. Register 2nd Authentication

1. Library Initialization

```
AuthLib mAuthLib = new AuthLib(context, API_2ND_AUTH_KEY);
```

2. Start Register 2nd Authentication

```
Intent intent = mAuthLib.signIn();
if (intent != null){
    startActivityForResult(intent, mAuthLib.EXCEL_KEY_ID_AUTH);
}else{
    // failure handling
}
```

3. Handle the Register result.

```
protected void onActivityResult(int requestCode, int resultCode, @Nullable Intent data) {
    super.onActivityResult(requestCode, resultCode, data);
    if (requestCode == mAuthLib.EXCEL_KEY_ID_AUTH) {
        if (resultCode == Activity.RESULT_OK) {
            String process = data.getStringExtra("process");
            if (mAuthLib.RESULT_REG.equals(process) ) {
                // Save AccessToken
                String AccessToken = data.getStringExtra("result");
            }
        }
    }
}
```

II. Start 2nd Authentication

1. Library Initialization

```
AuthLib mAuthLib = new AuthLib(context, API_2ND_AUTH_KEY);
```

2. Start 2nd Authentication

```
String AccessToken = "Saved AccessToken";
Intent intent = mAuthLib.signIn(AccessToken);
if (intent != null){
    startActivityForResult(intent, mAuthLib.EXCEL_KEY_ID_AUTH);
}else{
    // failure handling
}
```

3. Handle the Register result.

```
protected void onActivityResult(int requestCode, int resultCode, @Nullable Intent data) {
    super.onActivityResult(requestCode, resultCode, data);
    if (requestCode == mAuthLib.EXCEL_KEY_ID_AUTH) {
        if (resultCode == Activity.RESULT_OK) {
            String process = data.getStringExtra("process");
            if (mAuthLib.RESULT_AUTH.equals(process)) {
                // Verification from Server
                new Thread(new Runnable() {
                    @Override
                    public void run() {
                        if (mAuthLib.signCheck(result)) {
                            // success handling
                        } else {
                            // failure handling
                        }
                    }
                }).start();
            }
        }
    }
}
```

III. Withdraw 2nd Authentication

1. Library Initialization

```
AuthLib mAuthLib = new AuthLib(context, API_2ND_AUTH_KEY);
```

2. Start Withdraw 2nd Authentication

```
String AccessToken = "Saved AccessToken";  
Intent intent = mAuthLib.withdraw(AccessToken);  
if (intent != null){  
    startActivityForResult(intent, mAuthLib.EXCEL_KEY_ID_AUTH);  
}else{  
    // failure handling  
}
```

3. Handle the Register result.

```
protected void onActivityResult(int requestCode, int resultCode, @Nullable Intent data) {  
    super.onActivityResult(requestCode, resultCode, data);  
    if (requestCode == mAuthLib.EXCEL_KEY_ID_AUTH) {  
        if (resultCode == Activity.RESULT_OK) {  
            String process = data.getStringExtra("process");  
            if (mAuthLib.RESULT_WITHDRAW.equals(process)) {  
                // success handling  
            }  
        }  
    }  
}
```