Download

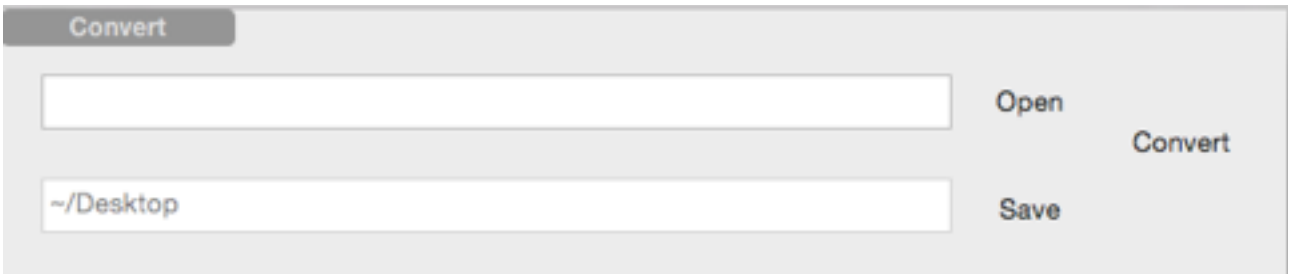
Link: http://pan.baidu.com/s/1kTrLWNH
password: 4tcq

Environment

mac os

Function Point

1.convert



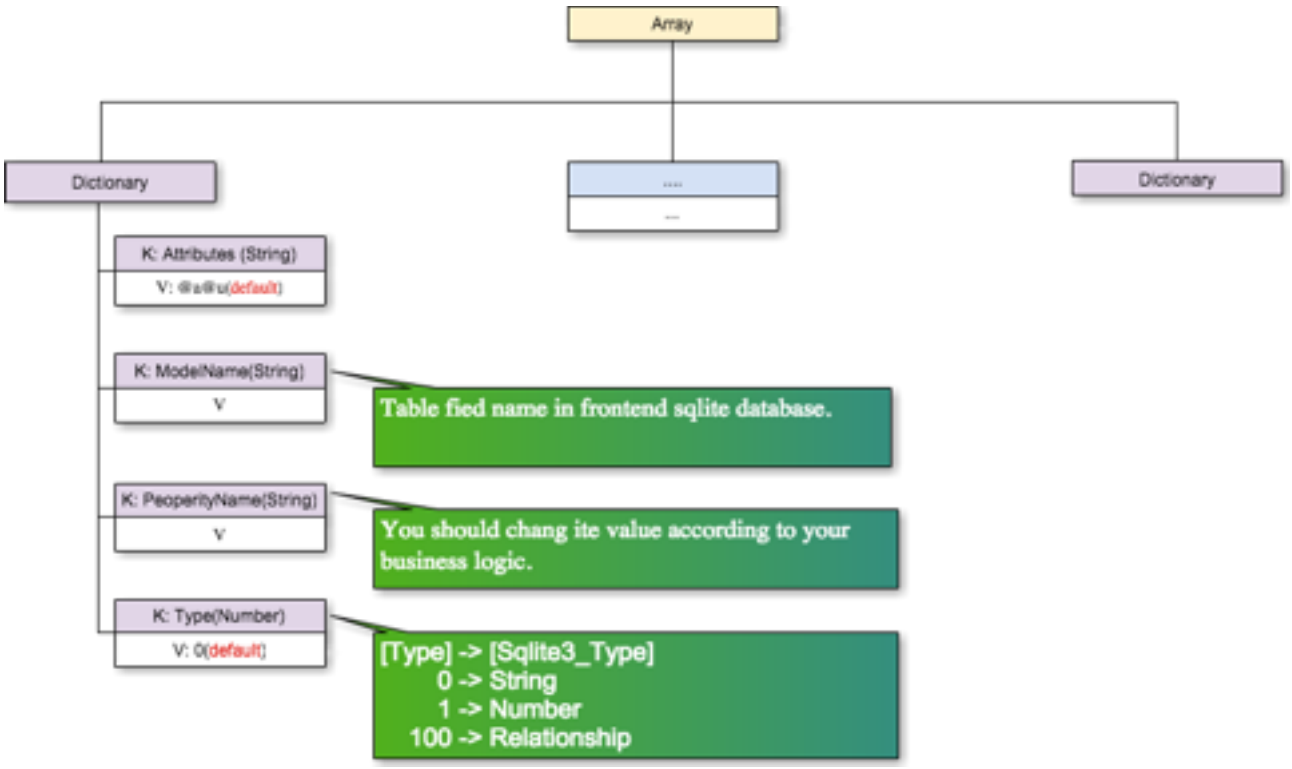
1.1 Brief

Convert every entity in CoreData file(.xcdatamodeld) to a corresponding plist file(.plist) with the same specific structure.

1.2 Description

- Open file extension should be .xcdatamodeld.
- Support drag and drop file to the text felid
- Each entity respectively generates a plist file.
- Plist file is named with [entityname] + .plist.

- The plist file structure is organised as:



- Sample:

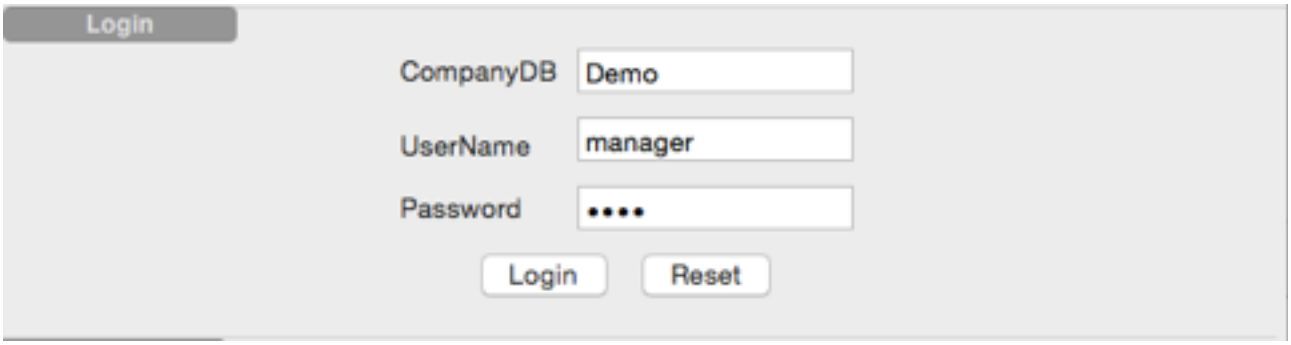
Key	Type	Value
▼ Root	Array	(3 items)
▼ Item 0	Dictionary	(4 items)
Attributes	String	@a@u
ModelName	String	id
PropertyName	String	Id
Type	Number	1
▼ Item 1	Dictionary	(4 items)
Attributes	String	@a@u
ModelName	String	password
PropertyName	String	Password
Type	Number	0
▼ Item 2	Dictionary	(4 items)
Attributes	String	@a@u
ModelName	String	user
PropertyName	String	User
Type	Number	0

1.3 Use Guide

- Open or drag .xcdatamodeld.

- Choose save file path or use the default path: ~/Desktop
- Click ‘Convert’. If it succeeds, click ‘OK’, it will open the save path in finder.

2.Login



1.1 Brief

Login to “https://10.58.114.44:40000/B1MobileServer/Login” with input text: ’CompanyDB’、‘’UserName’、’Password’.

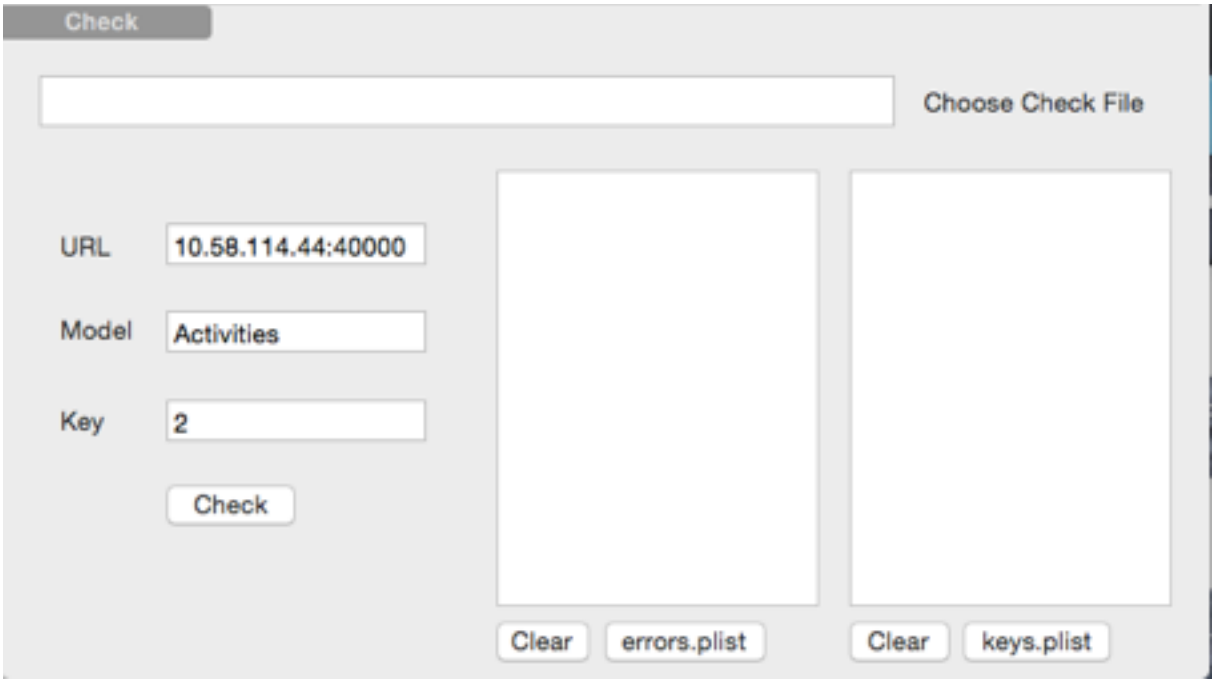
1.2 Description

- Login progress skip ssl.
- When login fail it will show HTTP response statusCode.

1.3 Use Guide

- Input required message.
- Click ‘Login’.

3.Check



1 Brief

After login, send the HTTP request to “https://[URL]/B1MobileServer/[Model]$key=[Key]”.And check all PropertyName in plist file whether it is in the backend table field from HTTP response whether .

1.2 Description

- In the destination, URL、Model、Key are based on your input. It may be like this:”https://10.58.114.44:40000/B1MobileServer/Activities?$key=2”
- The structure of check plist file must be same as the structure like the output plist in 1.Convert.
- After checking, the first error log display view on the [error.plist] button will display the error PropertityName to help user find the errors when typing.
- After checking, the second key log display view on the [keys.plist] button will display the keys from the HTTP response to help user to check.
- When errors are too many to find, you can click [errors.plist] button, it will generate a (entityname)_errors.plist under the same folder of check file to handle a great number of errors. It may be like this:

Key	Type	Value
▼ Root	Dictionary	(3 items)
Error_PropertyName_1	String	Id
Error_PropertyName_2	String	Password
Error_PropertyName_3	String	User

- When keys are too many to check, you can click [keys.plist] button, it will generate a (entityname)_keys.plsit under the same folder of check file to handle a great number of keys. It may be like this:

Key	Type	Value
▼ Root	Dictionary	(9 items)
Standard_PropertyName_1	String	DocType
Standard_PropertyName_2	String	Details
Standard_PropertyName_3	String	ActivityType
Standard_PropertyName_4	String	Closed
Standard_PropertyName_5	String	ActivityTime
Standard_PropertyName_6	String	HandledBy
Standard_PropertyName_7	String	CardCode
Standard_PropertyName_8	String	ContactPersonCode
Standard_PropertyName_9	String	AddressType

1.3 Use Guide

- Choose the plist file to check
- Input required message.
- Click ‘Check’ button.
- Through looking at the errors and keys logs to see
- When errors are too many, click”errors.plist” to see errors_log.plist.
- When keys are too many, click”keys.plist” to see keys_log.plist.