Archived:A contact list for choosing number(s) in PySymbian

Archived: This article is **archived** because it is not considered relevant for third-party developers creating commercial solutions today. If you think this article is still relevant, let us know by adding the template {{ReviewForRemovalFromArchive|user=~~~~|*write your reason here*}}.

The article is believed to be still valid for the original topic scope.

This PySymbian script shows a contact list class for choosing phone number(s), like the native one when sending SMS.

```
import appuifw, contacts

class ContactList:

    'show a contact list for choosing phone number'

    def __init__(self, sortable = None):
        'initialize variables'
        self.__db = contacts.open()
        self.__contact = []
        self.__number = {}
        self.__load_contact(sortable)

    def __load_contact(self, sortable = None):
        'load contact list from database'
        for id in self.__db:
            name = self.__db[id].title
            mobile = self.__db[id].find('mobile_number')
            phone = self.__db[id].find('phone_number')
            number_tmp = []
            if mobile:
                number_tmp.extend([field.value for field in mobile])
            if phone:
                number_tmp.extend([field.value for field in phone])
            self.__number[name] = number_tmp
            self.__contact.append(name)
            if sortable:
                self.__contact.sort(sortable)

    def select_single(self):
        'return a string contains a single number'
        index = appuifw.selection_list(self.__contact, 1)
        if index == None:
            return None
        number_list = self.__number[self.__contact[index]]
        if not number_list:
            return None
        if len(number_list) > 1:
            index = appuifw.popup_menu(number_list)
            if index == None:
                return None
        else:
            index = 0
        return number_list[index]
```


(C) Copyright Nokia 2014. All rights reserved.

```
    def select_multi(self):
        'return numbers in a list'
        index = appuifw.multi_selection_list(self.__contact, 'checkbox', 1)
        if not index:
            return None
        multi_number = []
        for i in index:
            number_list = self.__number[self.__contact[i]]
            if not number_list:
                continue
            if len(number_list) > 1:
                ind = appuifw.popup_menu(number_list)
                if ind == None:
                    continue
            else:
                ind = 0
            multi_number.append(number_list[ind])
        if multi_number:
            return multi_number
        else:
            return None

# here is some test
test = ContactList(lambda x, y:cmp(y, x))
print test.select_single()
print test.select_multi()
```


(C) Copyright Nokia 2014. All rights reserved.