Report: Vcard

Main.py

- Show menu()
- get_Details()
- make_vcard()
- write_vcard()
- read_vcard()
- find_index_opti()
- find_Card()
- delete_db()
- vcard_to_dict()
- delete_info()
- params()
- edit_info()
- print_all()

show_menu():

This function displays the menu and gets a number as the input from the user which is assigned to its appropriate field and returns the number you selected.

```
def show_menu():
    '''Function to show the menu options'''
    print()
    print("*************************")
    print(
         ' 1. Enter new vcard \n 2. Search Vcard \n 3. Print all the cards \n 4. Delete card \n 5. End Program')
    while True:
        print()
        num = input('Please select your choice: ')
        try:
            num = int(num)
        except Exception:
            print("Please enter integer values only from 1-6 and Try again !!!")
            continue
        if num >= 1 and num <= 6:
            break
        else:
            print("Please enter number from 1-6 only. Try again !!!")
    return num
```

get_Details():

This function prompts you to enter the required details

- Firstname
- Lastname
- Gender
- Email
- Phone Number
- Address
- Birthdate

And it stores all of this information into a dictionary and returns it.

```
def get_Details():
    '''
        Allowing the user to enter the vcard details
        All the possbile exceptions are handled in this
    '''
    print('Please enter the contact details  '.center(60, '='))
    details = {}
    details['first_name'] = input(' - First name            : ')
    while details['first_name'].isdigit():
        print('Please enter only characters for first name')
        details['first_name'] = input(' - First name          : ')
    details['last_name'] = input(' - Last name             : ')
    while details['last_name'].isdigit():
        print('Please enter only characters for last name')
        details['last_name'] = input(' - First name          : ')
    details['full_name'] = details['first_name'] + ' ' + details['last_name']
    details['email'] = input(' - E-mail address        : ')
    while "@" not in details['email']:
        print('Please enter valid email')
        details['email'] = input(' - E-mail address        : ')

    while True:
        try:
            details['phone_number'] = input(' - Phone number          : ')
            int(details['phone_number'])
            break
        except Exception:
            print('please enter only digits in phone number')
            continue
    details['vcard'] = make_vcard(**details)
    print(details['vcard'])
    return details

    details['vcf_file'] = f"{details['first_name'].lower() + details['last_name'].lower()}.vcf"

    return details
```

Make_vcard():

This function takes the vcard fields as parameters and makes it into the vcard format as well as returns it.

```
def make_vcard(
        first_name,
        last_name,
        full_name,
        email,
        phone_number,
        ):
    '''Function to convert the information in vcard format specified.'''
    return f"""
        BEGIN:vcard
        FN:{full_name}
        N:{last_name};{first_name}
        EMAIL;INTERNET:{email}
        TEL;WORK:{phone_number}
        VERSION: 2.1
        END:VCARD
    """
```

Write_vcard():

This function takes the details from the get_Details() function as a parameter and writes it to the vcf_database text file.

```
def write_vcard(details, fname='vcf_database.txt'):
    '''Function to write information in txt file'''
    with open(fname, 'a') as f:
        f.write(str(details['vcard']))
        f.write('\n')
        f.write('*' * 10)
```

Read_vcard():

This function takes a database and splits the database at the specified delimiter into separate vcards as well as stores and returns it in a list.

```
def read_vcard(fname='vcf_database.txt'):
    '''Function to read vcard information.'''
    delimiter = '\n' + '*' * 10
    with open(fname, 'r') as f:
        vcard = f.read().split(delimiter)
    return vcard
```

Find_index_opti():

This function reads the vcard which loops through the vcard and changes it into lower case characters -> converts into a dictionary and gets the index value as well as returns the vcard and the index .

```
def find_index_opti(key):
    '''
        Function to get the index for the given key
        Returns -1 incase key not found
    '''
    vcard = read_write_into_file.read_vcard()
    index = -1
    for i, v in enumerate(vcard):
        if v != '':
            key = key.lower()
            vc = vcard_to_dict(v)
            print(vc)
            if (key == vc['full_name'].lower()) or\
                    (key == vc['email'].lower()) or\
                    (key == vc['phone_number']):
                index = i
                break
    return vcard, index
```

find_Card()

This function uses find_index_opti() and gets vcard as well as the index.

If the record exists it prints the vcard and if it doesn't exist it prints 'Record Not Found'.

```
def find_Card(key):
    '''Function to find card for the given key.'''
    vcard, found = find_index_opti(key)
    if found != -1:
        print("Below details are found in the database")
        print(vcard[found], end='\n')
    else:
        print(f'Record not found for key {key}')
    return found
```

Delete_db():

This function deletes the vcard database (vcf_database.txt).

```
def delete_db(fname='vcf_database.txt'):
    '''Function to delete vcard information from txt file.'''
    try:
        os.unlink(fname)
    except Exception as e:
        print('database already deleted')
        print(e)
```

Vcard_to_dict():

This function takes the vcard as the parameter and loops through each line and splits items according to the format specified and appends values to another dictionary(vcard_list) as well as returns it.

```
def vcard_to_dict(vcard):
    '''
        Convert the vcard that is read from the text file to dict
        This make the search operation easy
    '''
    vcard_dict = {}
    vcard_list = []
    for i in vcard.split('\n'):
        if i.strip() != '':
            vcard_list.append(i)
    name = vcard_list[1].strip().replace('FN:', '').split()
    vcard_dict['first_name'] = name[0]
    vcard_dict['last_name'] = name[1]
    vcard_dict['full_name'] = name[0] + ' ' + name[1]
    vcard_dict['email'] = vcard_list[3].strip().replace('EMAIL;INTERNET:', '')
    vcard_dict['phone_number'] = vcard_list[4].strip().replace('TEL;WORK:', '')
    return vcard_dict
```

Delete_info():

This function gets the key and uses the find_index_opti function which returns the specified vcard as well as the index. It then deletes the database if a record exists and loops through the vcard and stores the details in the dictionary and prints ‘Record has been deleted’.

```
def delete_info(key):
    '''Deleting the given vcard.'''
    vcard, found = find_index_opti(key)
    if found != -1:
        read_write_into_file.delete_db()
        vcard[found] = ''

        for v in vcard:
            if v != '':
                details = {'vcard': v}
                read_write_into_file.write_vcard(details)

        print('Record has been deleted')

    return f'Record Not Found for given key {key}'
```

params():

This function is used in the edit_info function as it has a list of all the fields and prints the fields along with the number. After the user inputs the number it then indexes the appropriate field and returns it.

```
def params():
    cols = ['first_name', 'last_name', 'email', 'phone_number']
    for ind, col in enumerate(cols):
        print(f'{ind + 1} : {col}')
    number = int(input('Choose the number you want to edit : ')) - 1
    while number < 0 or number >= len(cols):
        number = int(input('Choose the number you want to edit : ')) - 1
    return cols[number]
```

Edit_info():

This function uses the find_index_opti function and if the index is greater than -1 it prints the vcard and asks you to input the number assigned to the field which you want to change and it also asks what you want to change it to . It then converts the vcard into the dictionary using vcard to the dictionary function and writes the edited info into the dictionary and if you have changed your name it also changes the full name, later it makes a new vcard with the updated information and stores it in the dictionary. It then deletes the database, loops through the created vcard and appends it to the parent dictionary which in turn is passed into the write_vcard function as writes the vcard to the text file.

```
def edit_info(key):
    '''Function responsible to make changes in the given vcard.'''
    vcard, found = find_index_opti(key)
    if found != -1:
        print(vcard[found], end='\n')
        From = params()
        To = input('Enter new information: ')
        vc = vcard_to_dict(vcard[found])
        vc[From] = To
        vc['full_name'] = vc['first_name'] + ' ' + vc['last_name']
        vc['vcard'] = make_vcard(**vc)
        vcard[found] = vc['vcard']
        read_write_into_file.delete_db()
        for v in vcard:
            if v != '':
                details = {'vcard': v}
                read_write_into_file.write_vcard(details)
    else:
        print('Record not found')
```

Print_all():

This function prints all the vcards in the database.

```
def print_all():
    '''Priting all the vcard information.'''
    vcard = read_write_into_file.read_vcard()
    for vc in vcard:
        if vc == "":
            continue
        print(vc, end='\n')
        print('--------------')
        print()
```