

Laboratorium Dasar

Fakultas Ilmu Komputer



Latihan Lab-13

Kasus-Buku Telepon!

Deskripsi: Pernahkah kamu melihat buku telepon? Buku yang berisi kumpulan nama, alamat, dan list satu atau lebih nomer telepon. Contohnya seperti ini:

Adi Alamat:Jln Melati NoTelp: 0982929292

Bunga Alamat:Jln Kembang NoTelp: 03213456

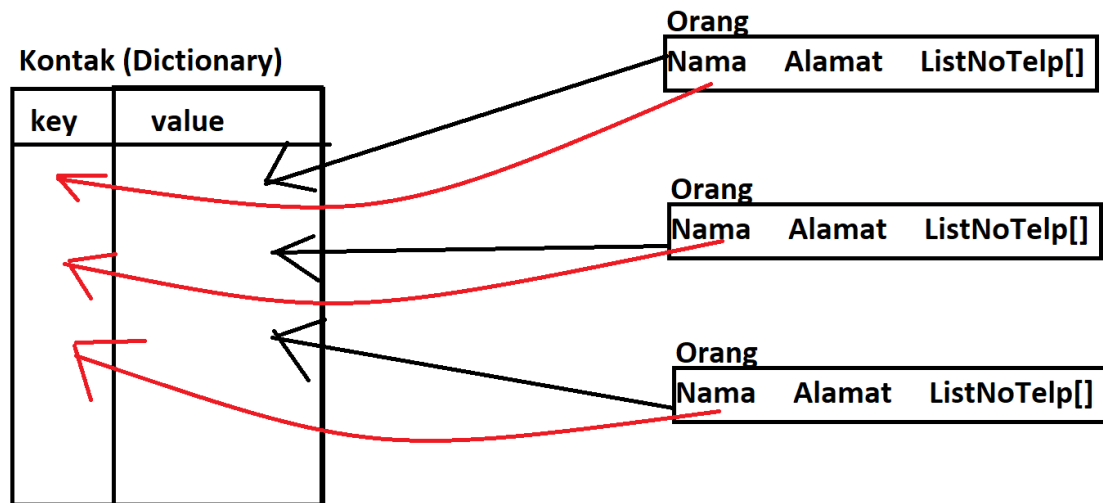
Nisa Alamat:Jln Mawar NoTelp: 0123123

Nisa Alamat:Jln Mawar NoTelp: 02112312

Lalu apakah kita bisa membuat program untuk mengisi atau menambah seseorang? Atau bahkan menghapus seseorang di daftar buku telepon kita? Tentu bisa. Kita bisa membuat representasi class dari setiap orang dahulu. Pada kasus ini kita bisa membuat class Orang dengan setidaknya memiliki data nama, alamat, dan list nomer telepon. Perhatikan bahwa nomer telepon ini bisa di isi juga dengan nomer handphone. Lalu kira-kira apa tipe data yang cocok untuk nama, alamat, dan list nomer telepon? Untuk nama dan alamat pasti kamu sudah tau, tapi bagaimana untuk list nomer telepon? Tentu saja List, namun list apakah? Integer atau String? Perhatikan bahwa integer tidak bisa diawali dengan 0. Contoh: 03213456. Maka tipe apakah yang cocok untuk list nomer telepon? Benar! Jadi List dari (yang jelas bukan integer).

Selanjutnya apakah sudah cukup kita membuat struktur data untuk Orang? Masih belum. Kita juga butuh struktur data untuk Buku Telepon itu sendiri. Apakah jika kita membuat class BukuTelepon dapat berinteraksi dengan class Orang? Tentu bisa! Mari kita lihat ilustrasi Gambar 1.

Perhatikan bahwa di class BukuTelepon kita bisa membuat dictionary sebut saja kontak. Kita bisa mengisi setiap elemen dictionary dengan objek Orang. Nama dari Objek orang bisa kita jadikan sebagai key/kunci. Dan value-nya adalah seluruh struktur data dari Objek orang tersebut. Dengan menempatkan nama sebagai key, maka kita juga dapat menggunakannya untuk menghapus elemen Kontak berdasarkan key nama tersebut.



Gambar 1. Ilustrasi Sistem

Baik, sekarang kalian sudah paham. Kesimpulannya kita akan membuat dua clas yaitu Orang dan BukuTelepon. Kemudian kita akan membuat kedua class tadi berinteraksi.

Input/Output Program

Penjelasan Input: -

Penjelasan Output: -

Bantuan Pseudocode/koding (mohon gunakan template ini):

```
class Orang(object):
    'Class Orang yang memuat data nama, alamat, dan list no telp'
    def __init__(self, nama, alamat, listNoTelp):
        # lakukan inisialisasi, assign nama, alamat, dan listNoTelp pada self
        # contoh:
        self.nama = nama
        # buat dua yang lain untuk self.alamat dan self.listNoTelp
        # MULAI (2 baris kode)

        # SELESAI

    def __str__(self):
        # Ini untuk memudahkan memasukkan value dengan
        # representasi semua string
        # buat string s untuk penampung
        s = ''
        # lakukan pengulangan pada item notelp didalam
```

```

        # listNoTelp yang sudah terurut
        for notelp in sorted(self.listNoTelp):
            # concat string s dengan string nama, alamat, dan notelp.
            # tab \t, baris baru \n
            s = s + self.nama+"\t Alamat:"+self.alamat+"\t NoTelp:"+notelp+'\n'

        # kembalikan s
        return s

class BukuTelepon(object):
    'Class BukuTelepon yang memuat kontak semua orang dan operasinya'
    def __init__(self):
        self.kontak = {} # dictionary kontak seluruh orang

    def tambah(self, nama, alamat, listNoTelp):
        # 1. lakukan buat objek p dengan instance dari class Orang
        # dengan inisialisasi nama, alamat, dan listNoTelp
        # 2. panggil self.kontak, isi key dengan nama dan value dengan p
        # lihat pada materi bagaimana cara menambahkan elemen pada dictionary
        # MULAI (2 baris kode)

        # SELESAI

    def hapus(self, nama):
        # hapus elemen dengan berdasarkan pada key
        # gunakan del dict[key], sesuaikan dict dengan key yang benar
        # MULAI (1 baris kode)

        # SELESAI

    def __str__(self):
        # Buat representasi string dari semua elemen dictionary
        # sehingga dapat dipanggil pada print(instanceObjek),
        # lihat bagian print(b)
        s = ''
        for p in sorted(self.kontak):
            s = s + str(self.kontak[p]) + '\n'
        return s

    def __call__(self, nama):
        # ini adalah special method yang akan di panggil dengan cara
        # instanceObjek('key'), lihat di main bagian b('Nisa')
        return self.kontak[nama]

```

```

def main():
    b = BukuTelepon()
    b.tambah('Bunga', 'Jln Kembang', ['03213456'])
    b.tambah('Nisa', 'Jln Mawar', ['0123123', '02112312'])
    b.tambah('Adi', 'Jln Melati', ['0982929292'])
    print("Data awal")
    print(b)
    setattr(b('Nisa'), 'alamat', 'Jln pelangi')
    print("Data Nisa saja setelah manipulasi")
    print(b('Nisa'))
    b.hapus("Bunga")
    print("Data setelah Bunga dihapus")
    print(b)

if __name__ == '__main__':
    main()

```

Contoh Output:

```

Data awal
Adi      Alamat:Jln Melati      NoTelp:0982929292

Bunga    Alamat:Jln Kembang     NoTelp:03213456

Nisa     Alamat:Jln Mawar       NoTelp:0123123
Nisa     Alamat:Jln Mawar       NoTelp:02112312

Data Nisa saja setelah manipulasi
Nisa     Alamat:Jln pelangi     NoTelp:0123123
Nisa     Alamat:Jln pelangi     NoTelp:02112312

Data setelah Bunga dihapus
Adi      Alamat:Jln Melati      NoTelp:0982929292

Nisa     Alamat:Jln pelangi     NoTelp:0123123
Nisa     Alamat:Jln pelangi     NoTelp:02112312

```