**LAPORAN PEMROGRAMAN BERORIENTASI OBJEK**

**TEMA: SOSIAL**

****

**Disusun oleh:**

**Raphael Hasiando Sihotang (5220411221)**

**PRORAM STUDI INFORMATIKA**

**FAKULTAS SAINS & TEKNOLOGI**

**UNIVERSITAS TEKNOLOGI YOGYAKARTA**

Jl. Ring Road Utara Jombor Sleman, Kab. Sleman, Prov. D.I. Yogyakarta –

Indonesia

2023

**DAFTAR ISI**

DAFTAR ISI 1

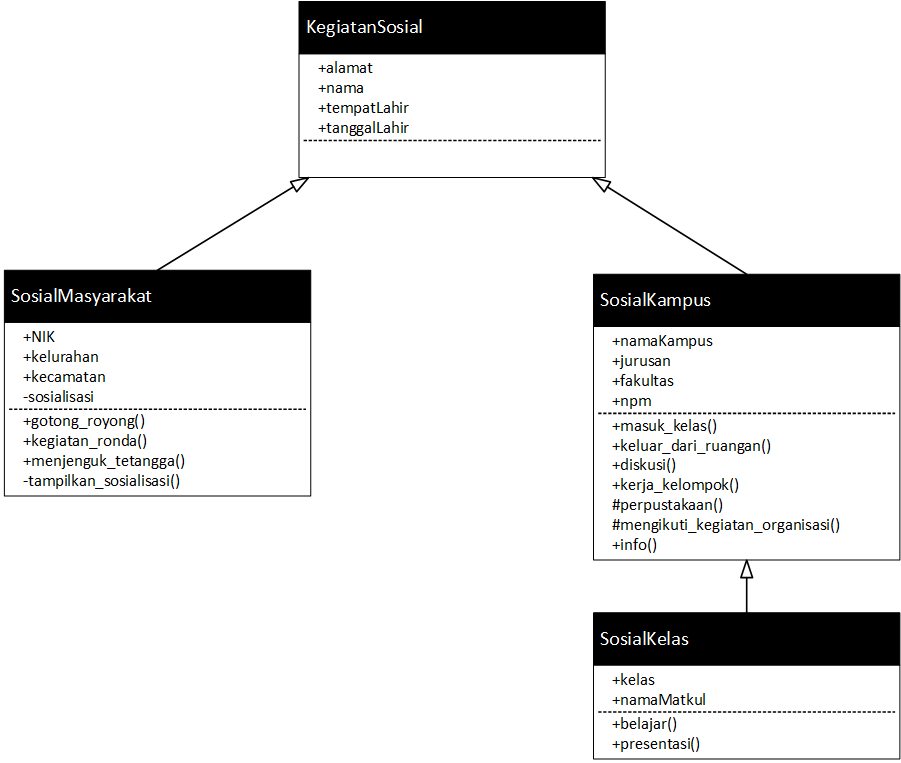
CLASS DIAGRAM 2

KODE PROGRAM 3

HASIL RUNNING PROGRAM 12

LINK GITHUB 24

CLASS DIAGRAM



KODE PROGRAM

#   Nama    :   Raphael Hasiando Sihotang

#   NPM     :   5220411221

#   Kelas   :   VI

#   Tema    :   Sosial

class KegiatanSosial:

    def \_\_init\_\_(self, nama, alamat, tempatLahir, tanggalLahir):

        self.nama = nama

        self.alamat = alamat

        self.tempatLahir = tempatLahir

        self.tanggalLahir = tanggalLahir

class SosialMasyarakat(KegiatanSosial):

    def \_\_init\_\_(self, nama, alamat, tempatLahir, tanggalLahir, NIK, kelurahan, kecamatan, sosialisasi):

        super().\_\_init\_\_(nama, alamat, tempatLahir, tanggalLahir)

        self.NIK = NIK

        self.kelurahan = kelurahan

        self.kecamatan = kecamatan

        # Akses Modifier

        self.\_\_sosialisasi = sosialisasi

    def gotong\_royong(self):

        print("Melakukan Gotong Royong \n")

        self.\_\_sosialisasi += 1

    def kegiatan\_ronda(self):

        print("Melakukan kegiatan Ronda malam \n")

        self.\_\_sosialisasi += 1

    def menjenguk\_tetangga(self):

        print("Menjenguk tetangga \n")

        self.\_\_sosialisasi += 1

    def \_\_tampilkan\_sosialisasi(self):

        if self.\_\_sosialisasi <= 3:

            print(f"Poin Sosialisasi anda masih kurang, Lakukanlah kegiatan lain supaya poin anda bertambah. \n")

        if self.\_\_sosialisasi > 3 and self.\_\_sosialisasi <= 6:

            print(f"Poin Sosialisasi anda mencapai {self.\_\_sosialisasi}, Tetaplah terus mengikuti kegiatan! \n")

        if self.\_\_sosialisasi > 6 and self.\_\_sosialisasi <= 10:

            print(f"Poin Sosialisasi anda mencapai {self.\_\_sosialisasi}. Anda telah mencapai tingkat yang terbaik! \n")

        if self.\_\_sosialisasi > 10:

            print(f"Poin Sosialisasi anda mencapai {self.\_\_sosialisasi}. Anda telah mencapai tingkat yang sangat terbaik! \n")

    def r\_tampilkan\_sosialisasi(self):

        self.\_\_tampilkan\_sosialisasi()

class SosialKampus(KegiatanSosial):

    def \_\_init\_\_(self, nama, alamat, tempatLahir, tanggalLahir, namaKampus, jurusan, fakultas, npm):

        super().\_\_init\_\_(nama, alamat, tempatLahir, tanggalLahir)

        self.namaKampus = namaKampus

        self.jurusan = jurusan

        self.fakultas = fakultas

        self.npm = npm

        #   Akses Modifier

        self.kondisiLokasi = False

        self.\_kondisiLokasi2 = False

        self.\_kondisiLokasi3 = False

    def masuk\_kelas(self):

        if self.kondisiLokasi is True:

            print("Anda sudah di dalam kelas...")

            return

        if self.\_kondisiLokasi2 is True:

            print("Mohon keluar dari perpustakaan...")

            return

        if self.\_kondisiLokasi3 is True:

            print("Mohon keluar dari ruangan organisasi...")

            return

        self.kondisiLokasi = True

        print("Memasuki kelas...")

    def keluar\_dari\_ruangan(self):

        if self.kondisiLokasi is True:

            print("Keluar dari kelas...")

            self.kondisiLokasi = False

            return

        if self.\_kondisiLokasi2 is True:

            print("Keluar dari perpustakaan...")

            self.\_kondisiLokasi2 = False

            return

        if self.\_kondisiLokasi3 is True:

            print("Keluar dari ruangan organisasi...")

            self.\_kondisiLokasi3 = False

            return

        print("Anda sudah di luar")

    def diskusi(self):

        if self.kondisiLokasi is True:

            print("Berdiskusi di dalam kelas...")

            return

        if self.\_kondisiLokasi2 is True:

            print("Berdiskusi di dalam perpustakaan...")

            return

        if self.\_kondisiLokasi3 is True:

            print("Berdiskusi di dalam ruangan organisasi...")

            return

        print("Berdiskusi di kampus")

    def kerja\_kelompok(self):

        if self.kondisiLokasi is True:

            print("Kerja Kelompok di dalam kelas...")

            return

        if self.\_kondisiLokasi2 is True:

            print("Kerja Kelompok di dalam perpustakaan...")

            return

        if self.\_kondisiLokasi3 is True:

            print("Kerja Kelompok di dalam ruangan organisasi...")

            return

        print("Kerja Kelompok di kampus...")

    #   Protected, kegiatan hanya dilakukan di luar kelas

    def \_perpustakaan(self):

        if self.kondisiLokasi is True:

            print("Mohon keluar dari kelas terlebih dahulu...")

            return

        if self.\_kondisiLokasi2 is True:

            print("Anda sudah di dalam perpustakaan..,")

            return

        if self.\_kondisiLokasi3 is True:

            print("Mohon keluar dari organisasi terlebih dahulu...")

            return

        print("Masuk ke perpustakaan...")

        self.\_kondisiLokasi2 = True

    #   Protected, kegiatan hanya dilakukan di luar kelas

    def \_mengikuti\_kegiatan\_organisasi(self):

        if self.kondisiLokasi is True:

            print("Mohon keluar dari kelas terlebih dahulu...")

            return

        if self.\_kondisiLokasi2 is True:

            print("Mohon keluar dari perpustakaan terlebih dahulu...")

            return

        if self.\_kondisiLokasi3 is True:

            print("Anda sudah di dalam ruangan organisasi...")

            return

        print("Masuk ke dalam ruangan organisasi...")

        self.\_kondisiLokasi3 = True

    def info(self):

        print(f"Nama            :   {self.nama}")

        print(f"Alamat          :   {self.alamat}")

        print(f"Tempat Lahir    :   {self.tempatLahir}")

        print(f"Tanggal Lahir   :   {self.tanggalLahir}")

        print(f"Nama Kampus     :   {self.namaKampus}")

        print(f"Jurusan         :   {self.jurusan}")

        print(f"Fakultas        :   {self.fakultas}")

        print(f"NPM             :   {self.npm}")

class SosialKelas(SosialKampus):

    def \_\_init\_\_(self, nama, alamat, tempatLahir, tanggalLahir, namaKampus, jurusan, fakultas, npm, kelas, namaMatkul):

        super().\_\_init\_\_(nama, alamat, tempatLahir, tanggalLahir, namaKampus, jurusan, fakultas, npm)

        self.kelas = kelas

        self.matkul = namaMatkul

    def belajar(self):

        if self.kondisiLokasi is False:

            print("Belajar di luar kelas...")

            return

        if self.kondisiLokasi is True:

            print("Belajar di dalam kelas")

    def presentasi(self):

        if self.kondisiLokasi is False:

            print("Harap Masuk ke dalam kelas terlebih dahulu")

            return

        print("Presentasi...")

    # Overriding, mengambil dan mengubah salah satu method dari kelas induknya

    def info(self):

        print(f"Nama            :   {self.nama}")

        print(f"Alamat          :   {self.alamat}")

        print(f"Tempat Lahir    :   {self.tempatLahir}")

        print(f"Tanggal Lahir   :   {self.tanggalLahir}")

        print(f"Nama Kampus     :   {self.namaKampus}")

        print(f"Jurusan         :   {self.jurusan}")

        print(f"Fakultas        :   {self.fakultas}")

        print(f"NPM             :   {self.npm}")

        print(f"Kelas           :   {self.kelas}")

        x = 0

        for i in self.matkul:

            x += 1

            print(f"Mata Kuliah {x}   :   {i}")

##################

### CONTROLLER ###

objects\_masyarakat = []       # Menyimpan objek ke dalam list kegiatan. Digunakan untuk mengakses salah satu dari beberapa objek

objects\_kampus = []

objects\_kelas = []

def main\_menu():

    CRUD()

    hasil\_input = sistem\_CRUD()

    if hasil\_input == 1:

        hasil\_input\_2 = kegiatan\_menu()

        if hasil\_input\_2 is True or hasil\_input\_2 is False or hasil\_input\_2 == 1:

            tambah\_data(hasil\_input\_2)

    if hasil\_input == 2:

        hasil\_input\_2 = kegiatan\_menu()

        if hasil\_input\_2 is True or hasil\_input\_2 is False or hasil\_input\_2 == 1:

            lihat\_data(hasil\_input\_2)

    if hasil\_input == 3:

        hasil\_input\_2 = kegiatan\_menu()

        if hasil\_input\_2 is True or hasil\_input\_2 is False or hasil\_input\_2 == 1:

            hapus\_data(hasil\_input\_2)

    if hasil\_input == 0:

        print("Terima Kasih")

        return

    main\_menu()

def CRUD():

    print("1.   Tambah Data")

    print("2.   Lihat Data")

    print("3.   Hapus Data")

    print("0.   Keluar")

def sistem\_CRUD():

    input\_user = int(input("==> "))

    if input\_user == 1:

        print("Tambah Data")

        return 1

    if input\_user == 2:

        print("Lihat Data")

        return 2

    if input\_user == 3:

        print("Hapus Data")

        return 3

    if input\_user == 0:

        print("Keluar dari sistem")

        return 0

    print("Input invalid, mohon diulangi")

    return sistem\_CRUD()

def kegiatan\_menu():

    tempat\_kegiatan()

    hasil\_input = input\_func()

    if hasil\_input is True or hasil\_input is False:

        if hasil\_input is True:

            return True

        if hasil\_input is False:

            return False

    else:

        if hasil\_input == 1:

            return 1

        print("Kembali ke main menu")

        return 0

def tempat\_kegiatan():

    print("Kegiatan di Lingkungan: ")

    print("1.   Masyarakat")

    print("2.   Kampus")

    print("3.   Kelas")

    print("0.   Kembali")

def input\_func():

    input\_user = int(input("==> "))

    if input\_user == 1:

        print("Masyarakat")

        return True

    if input\_user == 2:

        print("Kampus")

        return False

    if input\_user == 3:

        print("Kelas")

        return 1

    if input\_user == 0:

        return 0

    print("Input error, mohon diulangi")

    return input\_func()

def masyarakat(objek):

    print(f"Selamat Datang {objek.nama}")

    print("Sosial di lingkungan Masyarakat")

    print("1.   Gotong Royong")

    print("2.   Kegiatan Ronda malam")

    print("3.   Menjenguk Tetangga")

    print("4.   Tampilkan Sosialisasi anda")

    print("5.   Kembali ke main menu")

    input\_user = int(input("==> "))

    if input\_user == 1:

        objek.gotong\_royong()

        masyarakat(objek)

    elif input\_user == 2:

        objek.kegiatan\_ronda()

        masyarakat(objek)

    elif input\_user == 3:

        objek.menjenguk\_tetangga()

        masyarakat(objek)

    elif input\_user == 4:

        objek.r\_tampilkan\_sosialisasi()

        masyarakat(objek)

    elif input\_user == 5:

        print("Kembali ke main menu")

    else:

        print("Input Invalid, mohon diulangi \n")

        masyarakat(objek)

def kampus(objek):

    print(f"Selamat Datang {objek.nama}")

    print("Sosial di lingkungan Kampus")

    print("1.   Masuk Kelas")

    print("2.   Keluar dari ruangan")

    print("3.   Diskusi")

    print("4.   Kerja Kelompok")

    print("5.   Perpustakaan")

    print("6.   Mengikuti Kegiatan Organisasi")

    print("7.   Info Mahasiswa")

    print("8.   Kembali")

    input\_user = int(input("==> "))

    if input\_user == 1:

        objek.masuk\_kelas()

    if input\_user == 2:

        objek.keluar\_dari\_ruangan()

    if input\_user == 3:

        objek.diskusi()

    if input\_user == 4:

        objek.kerja\_kelompok()

    if input\_user == 5:

        objek.\_perpustakaan()

    if input\_user == 6:

        objek.\_mengikuti\_kegiatan\_organisasi()

    if input\_user == 7:

        objek.info()

    if input\_user == 8:

        print("Kembali")

        return

    kampus(objek)

def kelas(objek):

    print(f"Selamat Datang {objek.nama}")

    print("Sosial di lingkungan Kelas")

    print("1.   Masuk Kelas")

    print("2.   Keluar dari kelas")

    print("3.   Diskusi")

    print("4.   Kerja Kelompok")

    print("5.   Belajar")

    print("6.   Presentasi")

    print("7.   Info Mahasiswa")

    print("8.   Kembali")

    input\_user = int(input("==> "))

    if input\_user == 1:

        objek.masuk\_kelas()

    if input\_user == 2:

        objek.keluar\_dari\_ruangan()

    if input\_user == 3:

        objek.diskusi()

    if input\_user == 4:

        objek.kerja\_kelompok()

    if input\_user == 5:

        objek.belajar()

    if input\_user == 6:

        objek.presentasi()

    if input\_user == 7:

        objek.info()

    if input\_user == 8:

        print("Kembali")

        return

    kelas(objek)

def tambah\_data(kondisi):

    print("Silahkan isi data diri anda: ")

    hasil\_data = data\_diri()

    if kondisi is True or kondisi is False:

        if kondisi is True:

            hasil\_data\_masyarakat = data\_diri\_masyarakat()

            hdf = hasil\_data + hasil\_data\_masyarakat

            objek = SosialMasyarakat(hdf[0], hdf[1], hdf[2], hdf[3], hdf[4], hdf[5], hdf[6], 0)

            objects\_masyarakat.append(objek)

            masyarakat(objek)

        if kondisi is False:

            hasil\_data\_kampus = data\_diri\_kampus()

            hdf = hasil\_data + hasil\_data\_kampus

            objek = SosialKampus(hdf[0], hdf[1], hdf[2], hdf[3], hdf[4], hdf[5], hdf[6], hdf[7])

            objects\_kampus.append(objek)

            kampus(objek)

    else:

        if kondisi == 1:

            hasil\_data\_kampus = data\_diri\_kampus()

            hasil\_data\_kelas = data\_diri\_kelas()

            hdf = hasil\_data + hasil\_data\_kampus + hasil\_data\_kelas

            objek = SosialKelas(hdf[0], hdf[1], hdf[2], hdf[3], hdf[4], hdf[5], hdf[6], hdf[7], hdf[8], hdf[9])

            objects\_kelas.append(objek)

            kelas(objek)

def tampil\_data(kondisi):

    if kondisi is True or kondisi is False:

        if kondisi is True:

            x = 0

            for i in objects\_masyarakat:

                x += 1

                print(f"{x}.    {i.nama}    |   {i.NIK}")

            print("\n")

        if kondisi is False:

            x = 0

            for i in objects\_kampus:

                x += 1

                print(f"{x}.    {i.nama}    |   {i.npm}")

            print("\n")

    else:

        if kondisi == 1:

            x = 0

            for i in objects\_kelas:

                x += 1

                print(f"{x}.    {i.nama}    |   {i.npm}")

            print("\n")

def lihat\_data(kondisi):

    if kondisi is True or kondisi is False:

        if kondisi is True:

            objek = objects\_masyarakat

        if kondisi is False:

            objek = objects\_kampus

    else:

        if kondisi == 1:

            objek = objects\_kelas

    if len(objek) == 0:

        print("Data masih kosong, mohon menambahkan data terlebih dahulu")

        return

    tampil\_data(kondisi)

    print("Pilih salah satu angka dari data diatas yang akan dipakai")

    input\_user = int(input("==> "))

    if input\_user > 0 and input\_user <= len(objek):

        if kondisi is True or kondisi is False:

            if kondisi is True:

                masyarakat(objek[input\_user-1])

            if kondisi is False:

                kampus(objek[input\_user-1])

        else:

            if kondisi == 1:

                kelas(objek[input\_user-1])

def hapus\_data(kondisi):

    if kondisi is True or kondisi is False:

        if kondisi is True:

            objek = objects\_masyarakat

        if kondisi is False:

            objek = objects\_kampus

    else:

        if kondisi == 1:

            objek = objects\_kelas

    if len(objek) == 0:

        print("Data masih kosong, mohon menambahkan data terlebih dahulu")

        return

    tampil\_data(kondisi)

    print("Pilih salah satu angka dari data diatas yang akan dipakai")

    input\_user = int(input("==> "))

    if input\_user > 0 and input\_user <= len(objek):

        objek.pop(input\_user-1)

def data\_diri():

    nama = input("Nama               :")

    alamat = input("Alamat             :")

    tempatLahir = input("Tempat Lahir       :")

    tanggalLahir = input("Tanggal Lahir      :")

    return [nama, alamat, tempatLahir, tanggalLahir]

def data\_diri\_masyarakat():

    nik = input("NIK                :")

    kelurahan = input("Kelurahan          :")

    kecamatan = input("Kecamatan          :")

    return [nik, kelurahan, kecamatan]

def data\_diri\_kampus():

    namaKampus = input("Nama Kampus        :")

    jurusan = input("Jurusan            :")

    fakultas = input("Fakultas           :")

    npm = input("NPM                :")

    return [namaKampus, jurusan, fakultas, npm]

def data\_diri\_kelas():

    kelas = input("Kelas              :")

    matakuliah = []

    print("Berapa Matakuliah yang akan diambil? (Maksimal 5)")

    def matakuliah\_func():

        matakuliah\_qty = int(input("==> "))

        if matakuliah\_qty > 0 and matakuliah\_qty <= 5:

            for i in range(matakuliah\_qty):

                matakuliah.append(input(f"Matakuliah {i+1}       :"))

        else:

            print("Input invalid, mohon diulangi...")

            matakuliah\_func()

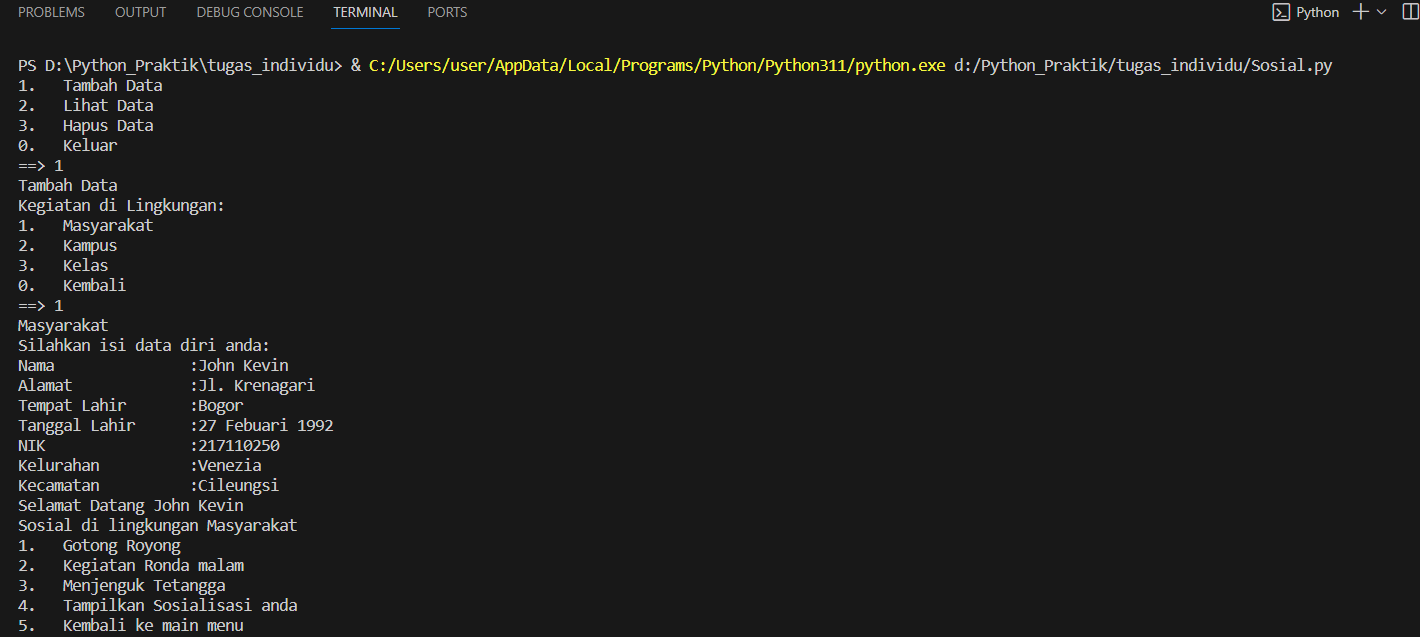
    matakuliah\_func()

    return [kelas, matakuliah]

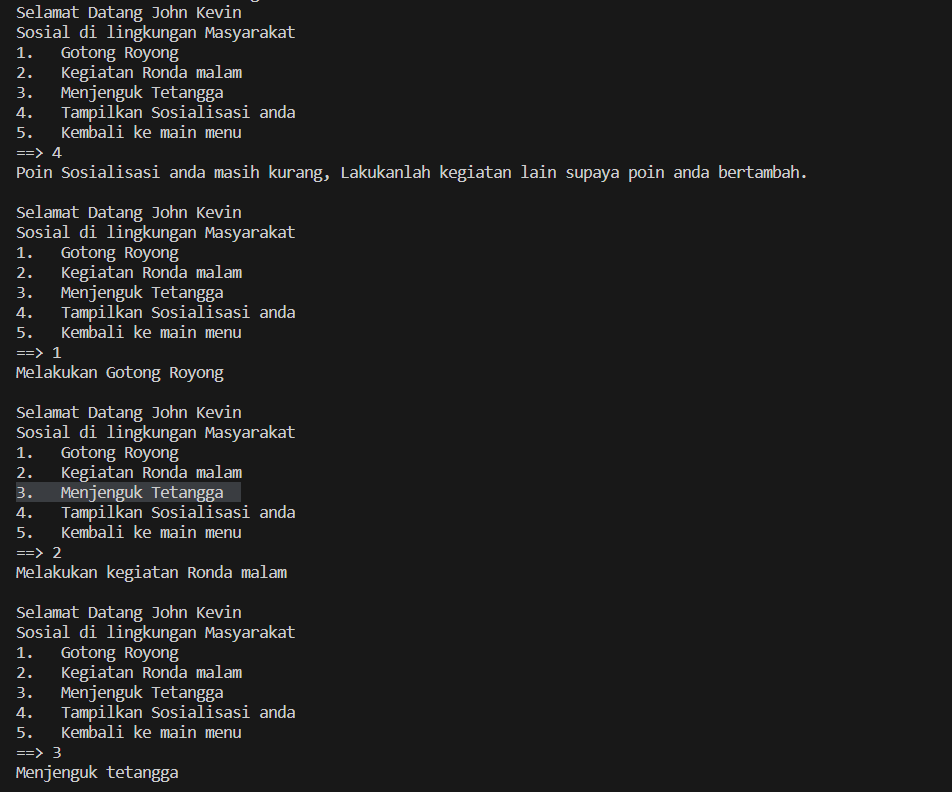
main\_menu()

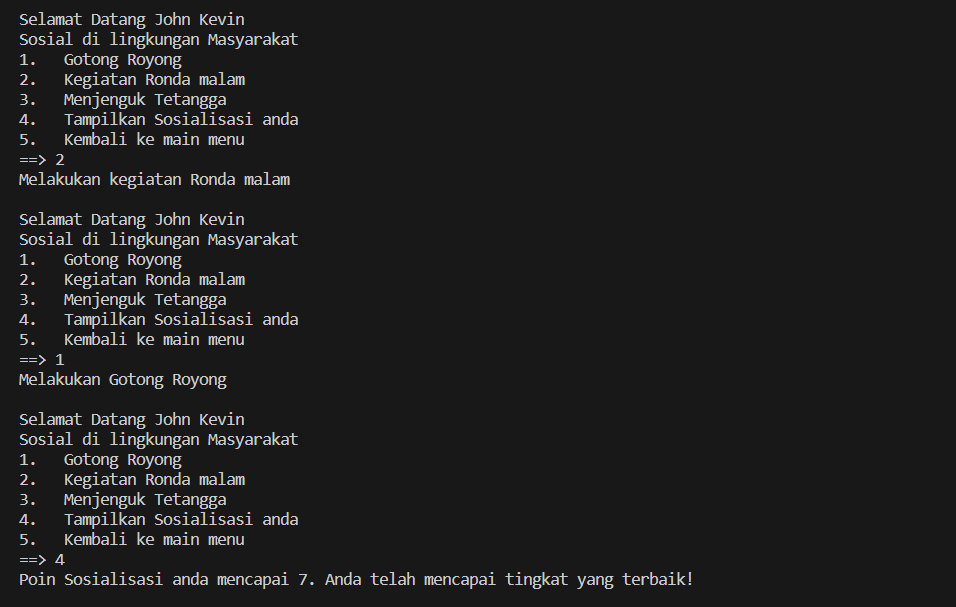
HASIL RUNNING PROGRAM

1. Menambahkan objek baru untuk kelas SosialMasyarakat. Artinya, menambahkan data baru di lingkungan masyarakat

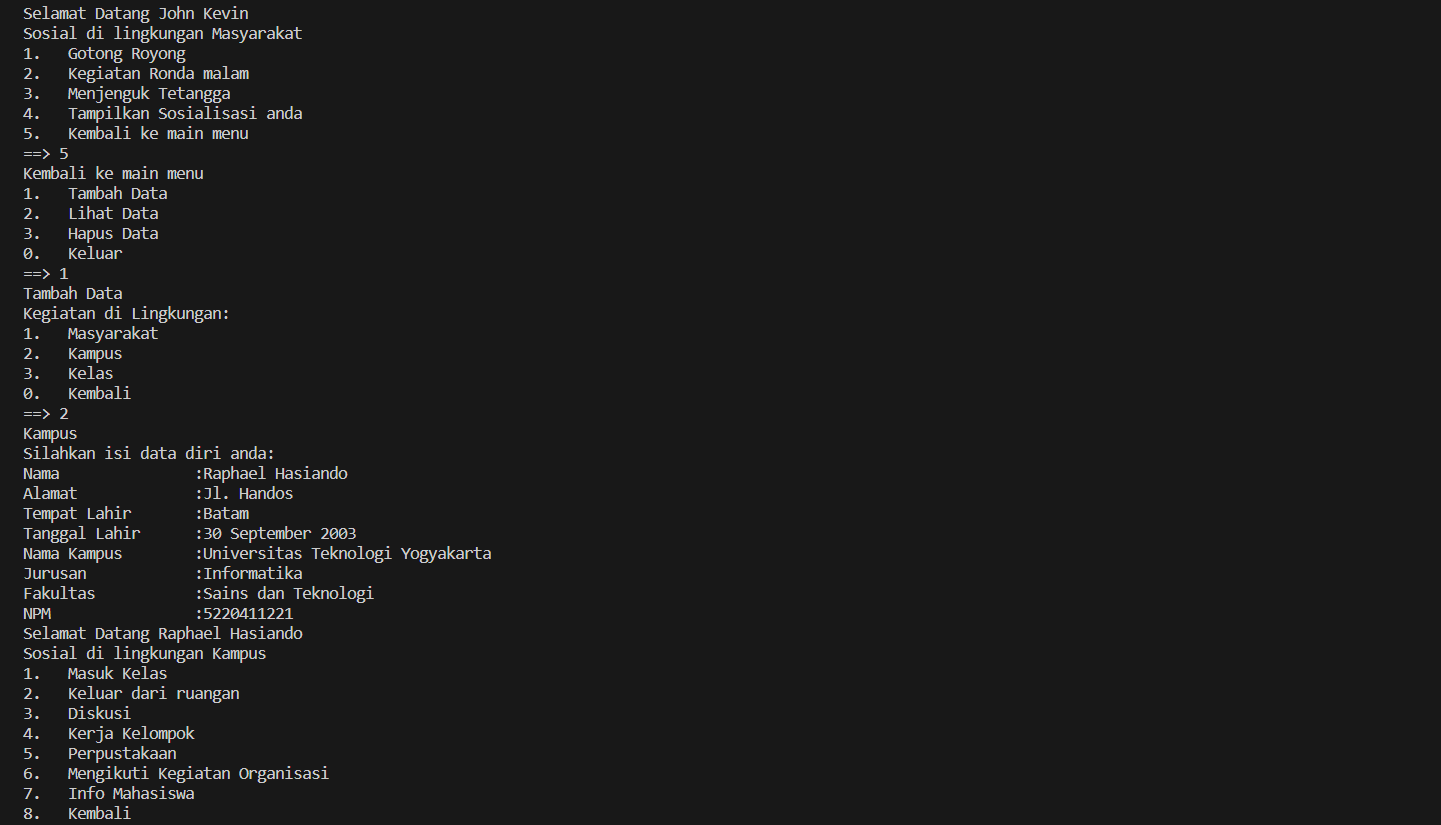


1. Melakukan Kegiatan sosial dengan cara memilih salah satu dari 3 kegiatan. Kegiatan akan dilakukan sebanyak 7 kali. Jika pengguna melakukan kegiatan, pengguna mendapatkan 1 poin sosialisasi.

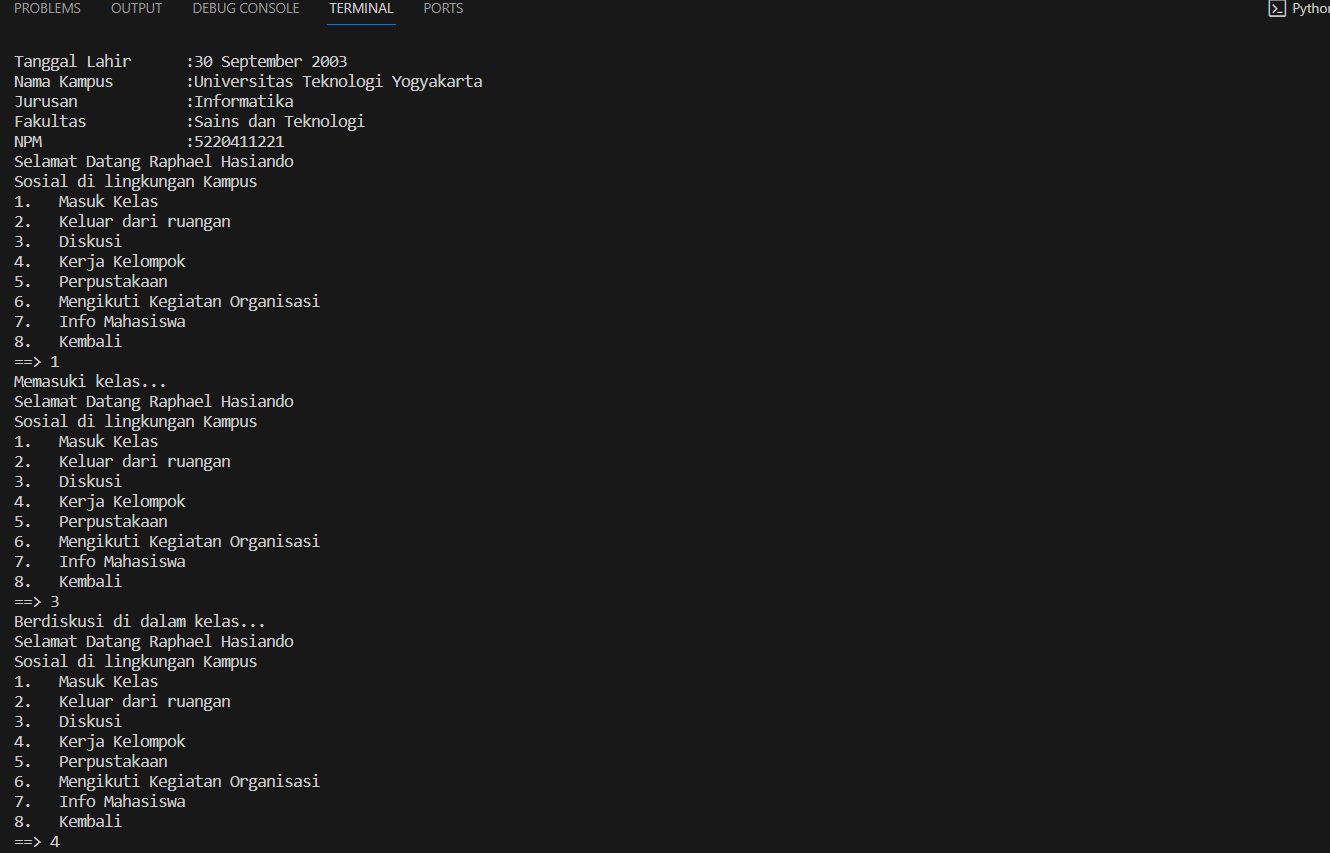


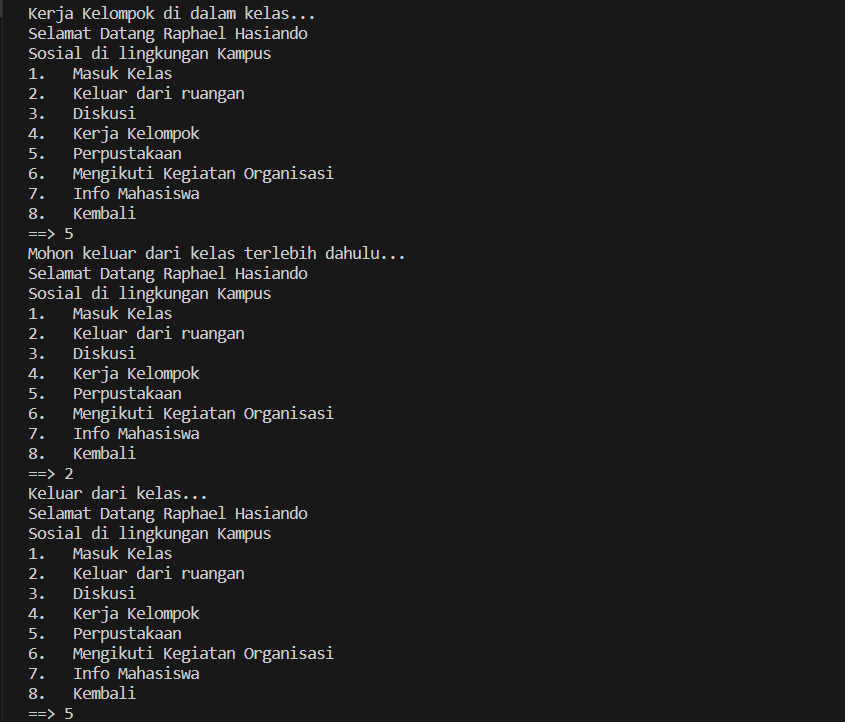


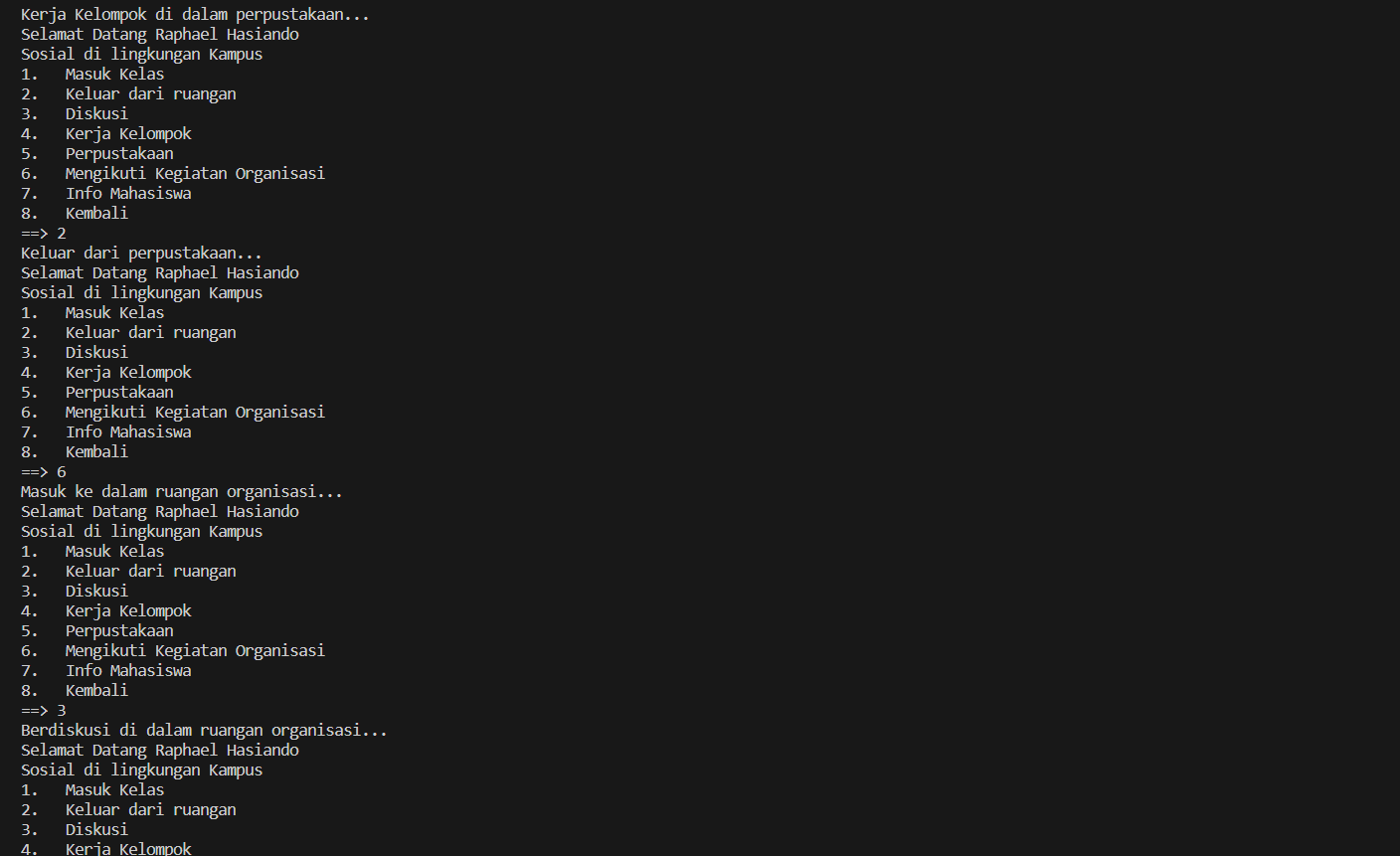
1. Menambahkan objek baru untuk kelas SosialKampus.

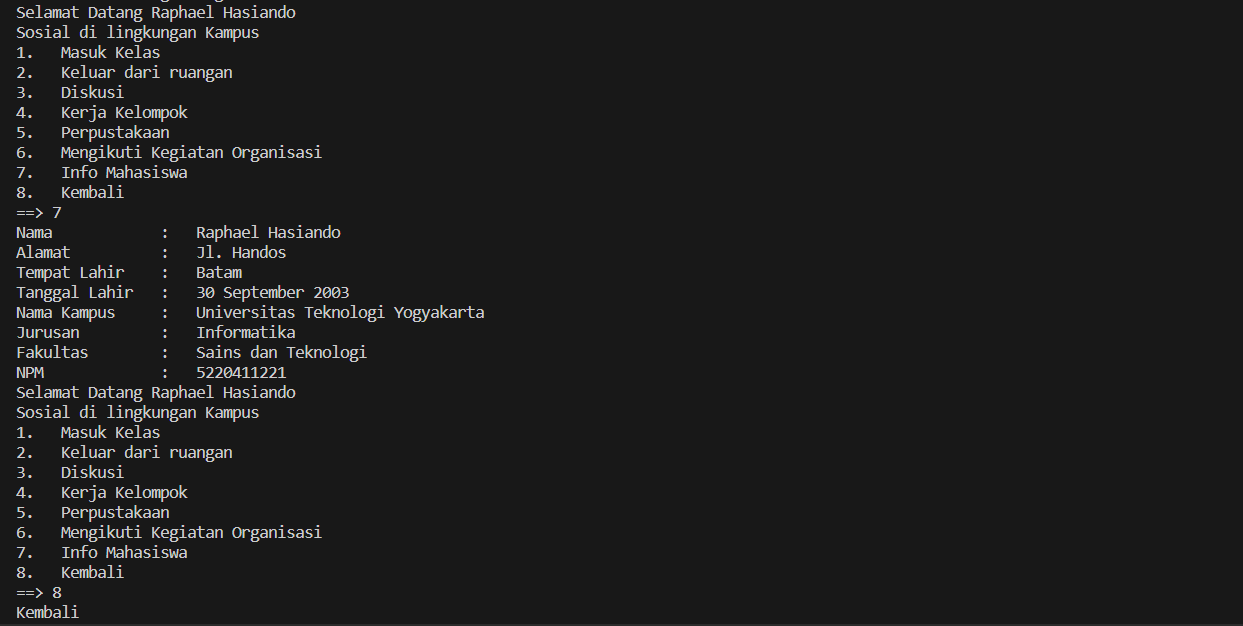


1. Melakukan kegiatan di kampus. Jika pengguna masuk ke dalam kelas, maka ia harus keluar dari ruangan terlebih dahulu jika ia ingin masuk ke perpustakaan atau ruangan organisasi. Jika Pengguna melakukan kegiatan sosialisasi seperti diskusi dan kerja kelompok di tempat yang berbeda, maka ia mendapatkan peringatan yang berbeda tergantung dimana ia berada.

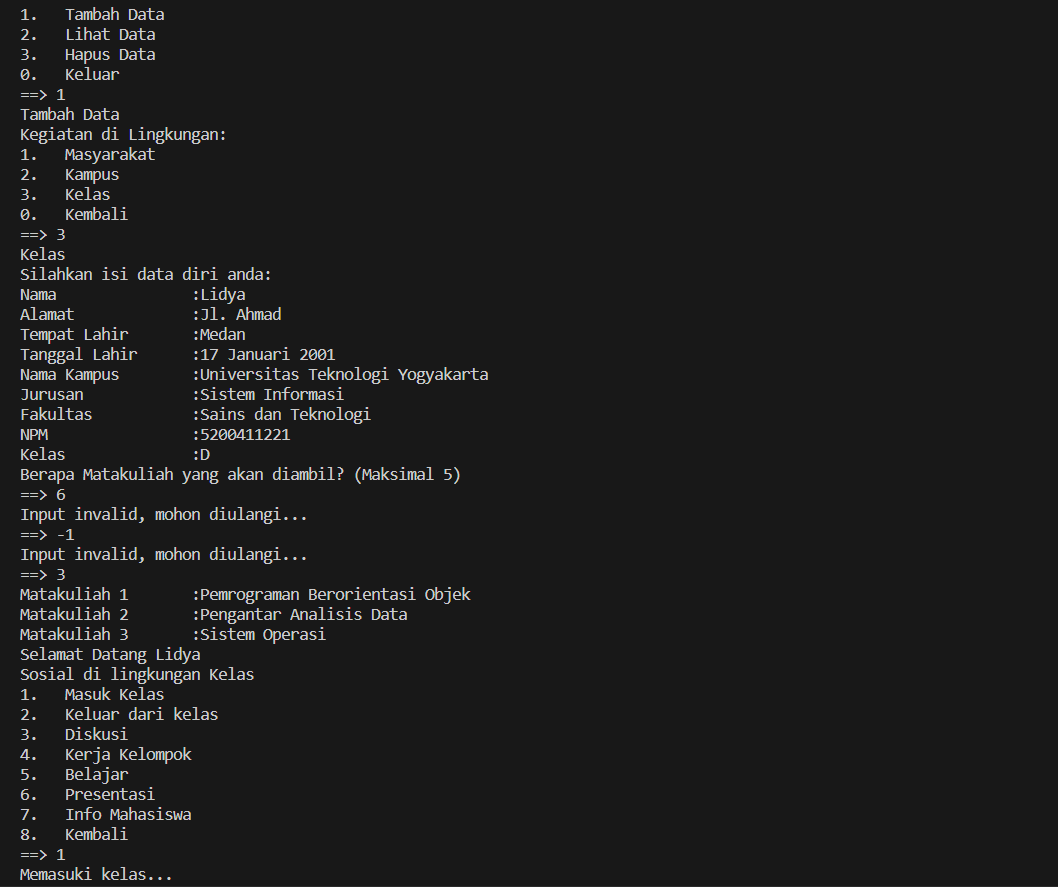


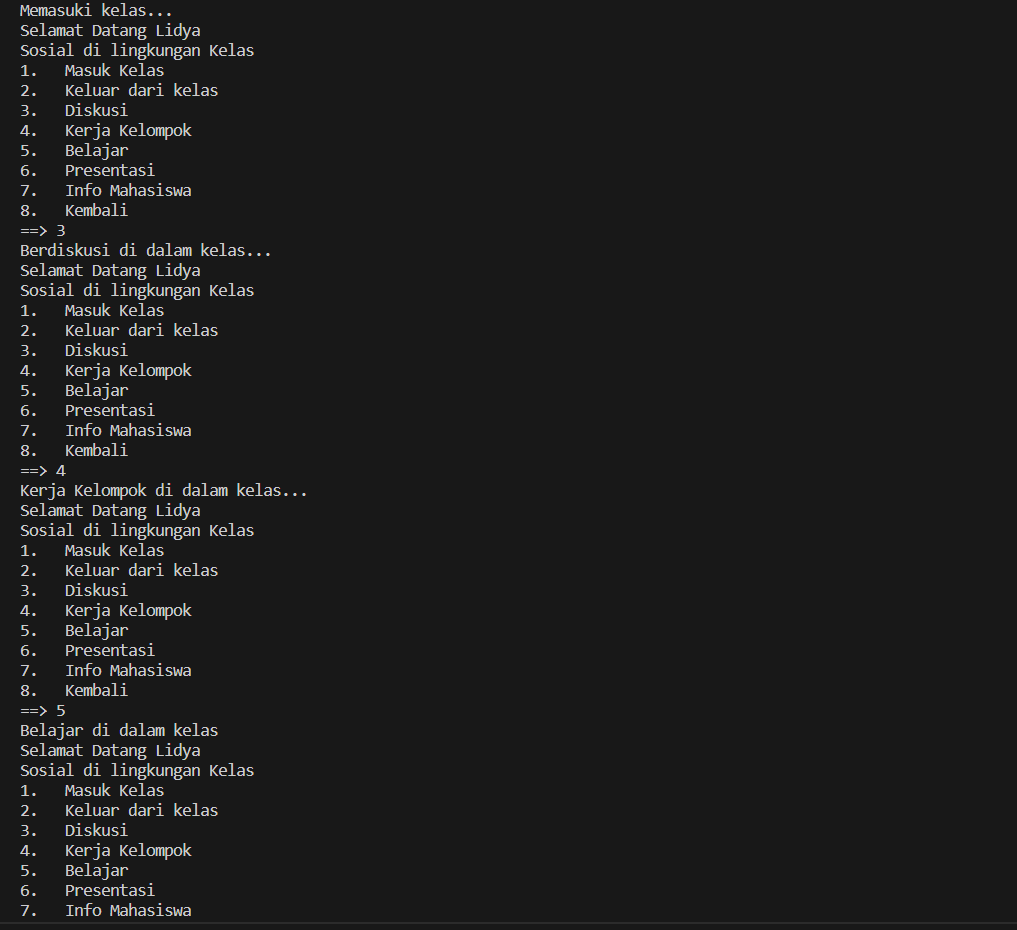


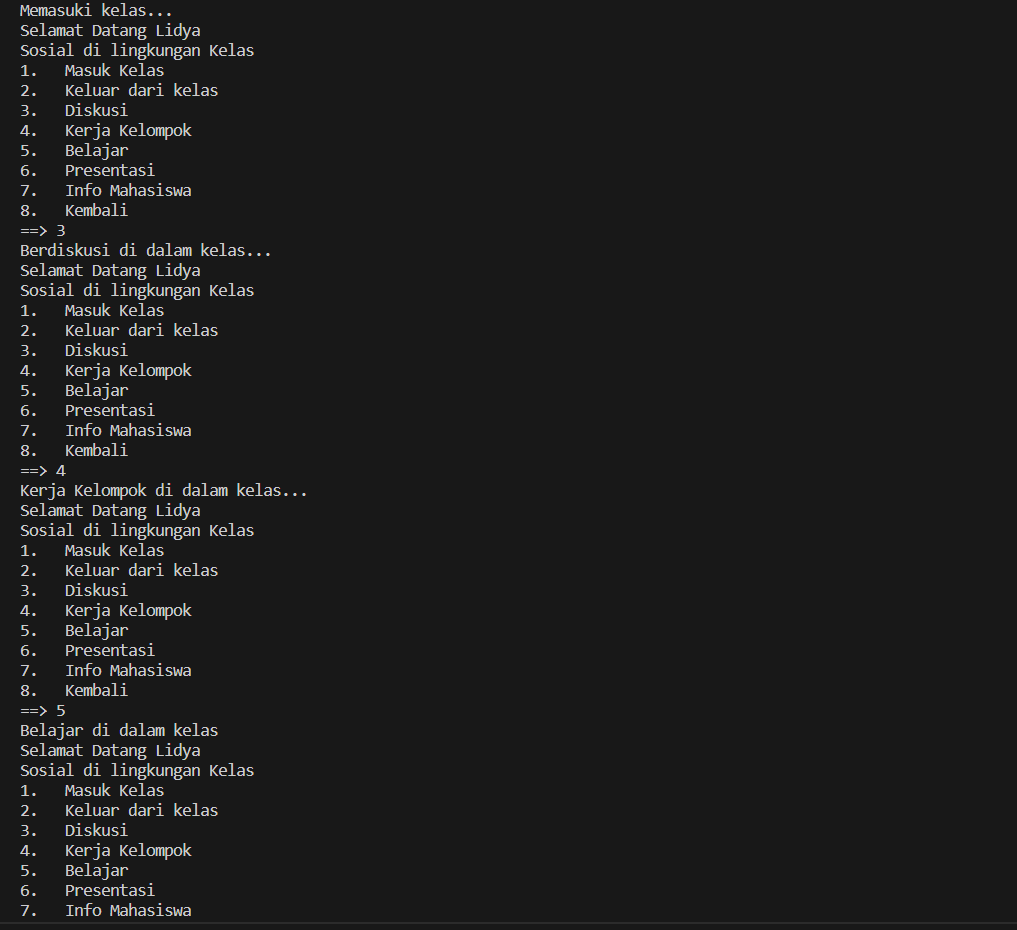


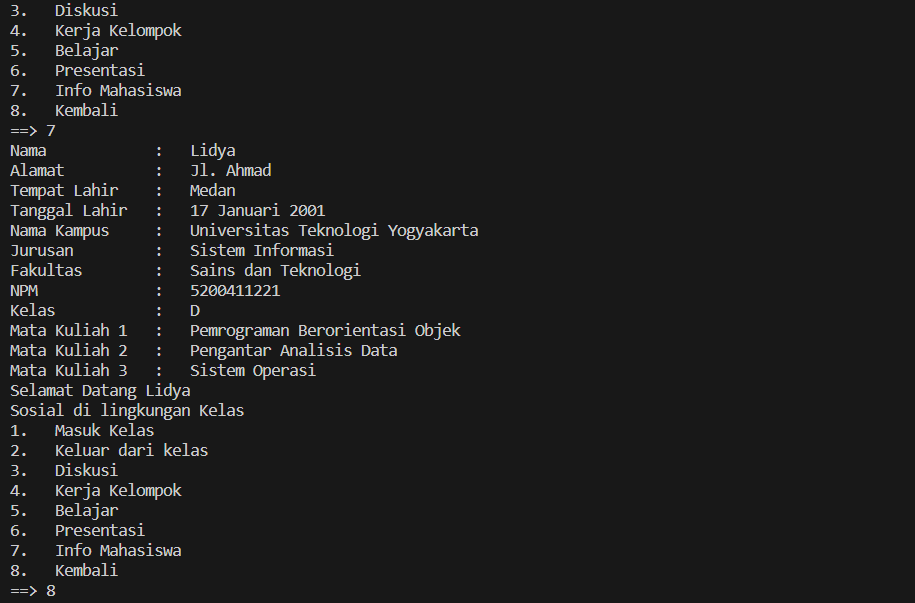


1. Menambahkan objek baru untuk kelas SosialKelas.

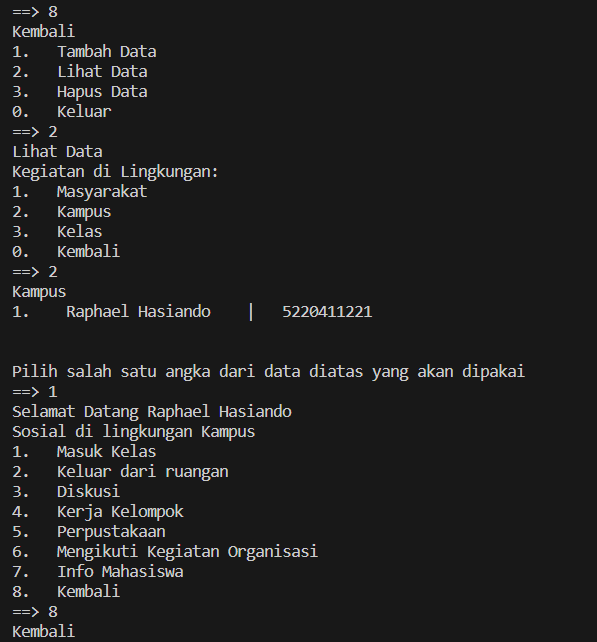




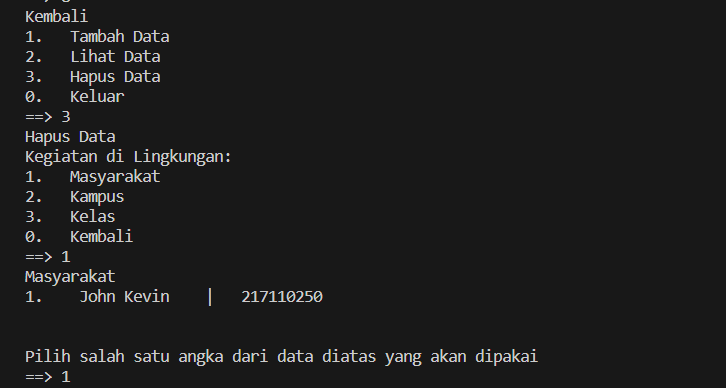


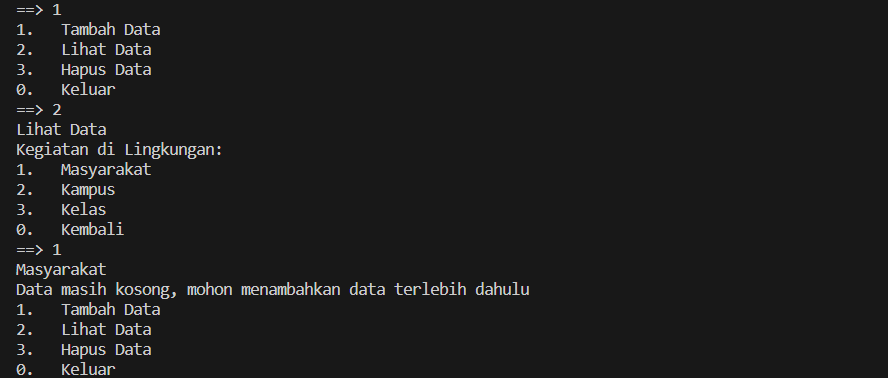


1. Menggunakan fitur lihat data, digunakan untuk mengakses pada objek tersebut yang sudah disimpan di dalam list.

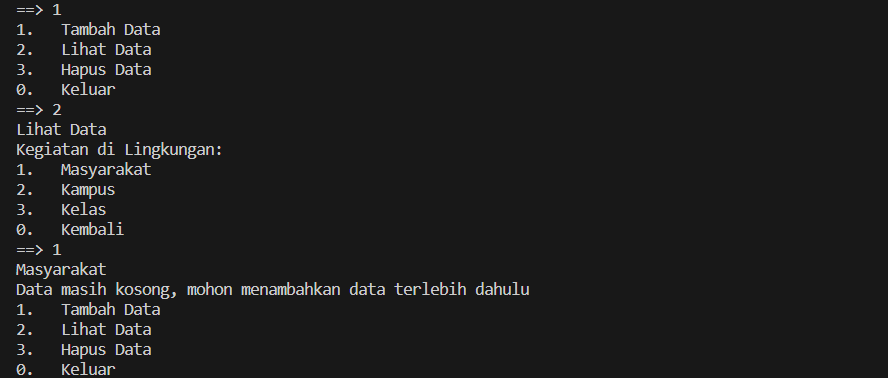


1. Menghapus objek salah satu objek di dalam list yang tersedia.

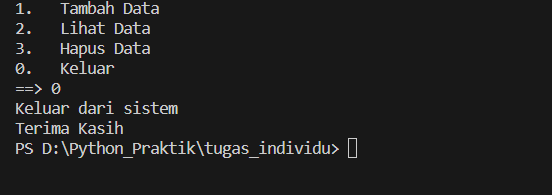




1. Memasukkan input yang tidak tersedia di list



1. Program Selesai



LINK GITHUB

<https://github.com/RaphaelHasiando/PBOVI_5220411221_Raphael-Hasiando-Sihotang/tree/7faf34af9cec3d208157a340d07b09b201b34a9e/week12>