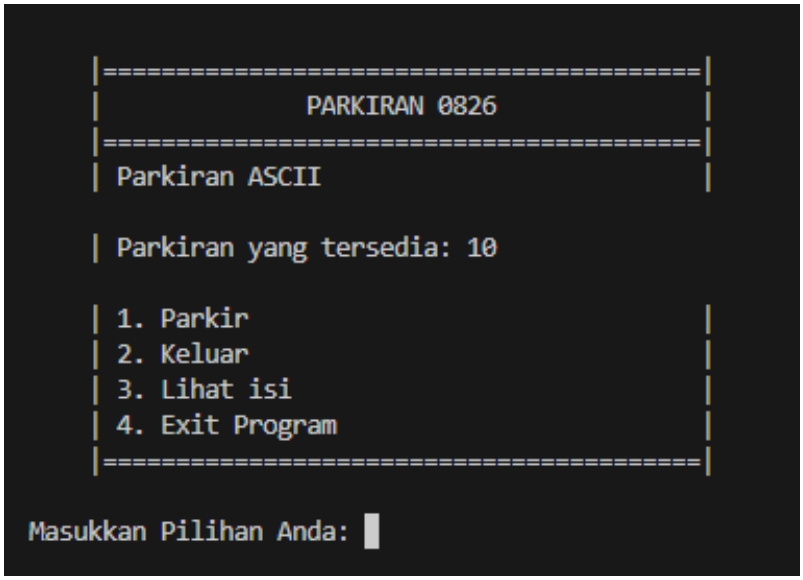
Nama : Danang Tri Atmaja

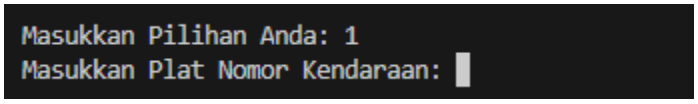
NIM : 22.83.0826

Kelas : 22-TK02

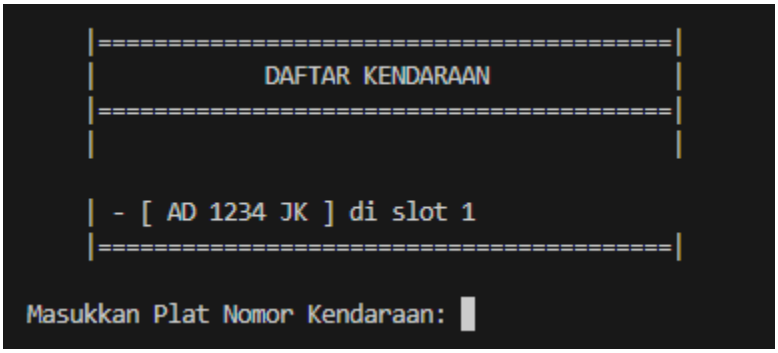
PROGRAM PYTHON PARKIRAN



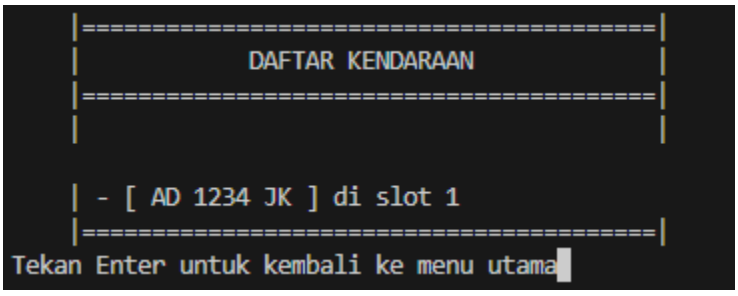
Home Screen



Pilihan 1



Pilihan 2



Pilihan 3

A. Penjelasan Program

Program ini ditujukan pada pintu masuk dan pintu keluar parkiran agar kendaraan tetap terdata dengan baik, ditambah dengan UI ASCII untuk meningkatkan user experience.

B. Source Code

```
import datetime
import time
import os
import numpy as np

max_parkiran = 10

parking_lot = np.zeros(max_parkiran, dtype=int) #numpy array lahan parkir
parked_vehicles = {}
log = {}
log_counter = 0

def main():
    while True:
        render_ui()
        choice = input("Masukkan Pilihan Anda: ")
        if choice == '1':
            park()
        elif choice == '2':
            render_parking_list()
            print()
            exit_parking()
        elif choice == '3':
            render_parking_list()
            input("Tekan Enter untuk kembali ke menu utama")
        elif choice == '4':
            exit_program()
        else:
            render_error("Inputan tidak Valid!")
            time.sleep(2)

def park():
    global max_parkiran
    if len(parked_vehicles) == max_parkiran:
        render_error("Parkiran Penuh!")
        time.sleep(2)
    else:
        plate_number = input("Masukkan Plat Nomor Kendaraan: ")
        print()
        plate_number = plate_number.upper()
        if plate_number in parked_vehicles:
            render_error("Kendaraan dengan Plat Nomor tersebut sudah terparkir!")
            time.sleep(2)
            return
        parking_slot = np.where(parking_lot == 0)[0][0]
```

```
        parking_lot[parking_slot] = 1
        parked_vehicles[plate_number] = parking_slot
        print("Kendaraan", plate_number, "telah diparkir di slot", parking_slot + 1)
        max_parkiran -=1
        time.sleep(2)

def exit_parking():
    global log_entry, max_parkiran
    if len(parked_vehicles) == 0:
        render_error("Tidak ada kendaraan yang terparkir di parkiran")
        time.sleep(2)
    else:
        plate_number = input("Masukkan Plat Nomor Kendaraan: ")
        print()
        plate_number = plate_number.upper()
        if plate_number not in parked_vehicles:
            render_error("Kendaraan tidak terparkir di parkiran")
            time.sleep(2)
            return
        parking_slot = parked_vehicles[plate_number]
        parking_lot[parking_slot] = 0
        del parked_vehicles[plate_number]
        print("Kendaraan", plate_number, "telah keluar dari parkiran")
        print()
        log_parking_event(plate_number)
        log_exit_event(plate_number)
        print('Biaya : Rp.',log_entry['Biaya'])
        print()
        input ("Tekan enter untuk lanjut.")
        max_parkiran +=1

def get_cost(enter_time, exit_time):
    parking_duration = exit_time - enter_time
    parking_hours = parking_duration.total_seconds() / 3600
    parking_rate_per_hour = 3000
    parking_rate_much = 5000
    if parking_hours <= 1:
        parking_cost = parking_rate_per_hour
    elif parking_hours > 1:
        parking_cost = parking_rate_much
    return parking_cost

def render_ui():
    clear_screen()
    menu = '''
    |=======================================|
    |              PARKIRAN 0826            |
    |=======================================|
    | Parkiran ASCII                        |
    '''
```

```
    print(menu)
    print("    | Parkiran yang tersedia:", max_parkiran)
    lanjut = '''
    | 1. Parkir                              |
    | 2. Keluar                              |
    | 3. Lihat isi                           |
    | 4. Exit Program                        |
    |========================================|
    '''
    print(lanjut)

def render_error(message):
    clear_screen()
    error = '''
    |========================================|
    |                 ERROR                  |
    |========================================|
    |      ______                            |
    |     |  ____|                           |
    |     | |__    _ __  _ __  ___   _ __    |
    |     |  __|  | '__|| '__/ _ \ | '__|    |
    |     | |____ | |   | | | (_) || |       |
    |     |______||_|   |_|  \___/ |_|       |
    |========================================|
    '''
    print(error)
    print(message)

def render_exit():
    clear_screen()
    exit = '''
    |========================================|
    |             TERIMA KASIH               |
    |========================================|
    |                                        |
    |    Terima kasih atas kunjungan Anda    |
    |========================================|
    '''
    print(exit)

def render_parking_list():#fungsi menampilkan list kendaraan yang terparkir
    clear_screen()
    list_park = '''
    |========================================|
    |            DAFTAR KENDARAAN            |
    |========================================|
    |                                        |
    '''
    print(list_park)
    if len(parked_vehicles) == 0:
```

```
        print("    |            Parkiran Kosong !            |")
    else:
        for plate_number, parking_slot in parked_vehicles.items():
            print("    | - [", plate_number, "] di slot", parking_slot + 1)
    print('    |=========================================|')


def exit_program(): #fungsi keluar program dan menampilkan data data yang telah tersimpan
    render_exit()
    time.sleep(3)
    clear_screen()
    for log_entry in log.values():
        print('Plat:', log_entry['Plat'],'=====> Rp.', log_entry['Biaya'])
        print()
    input("Tekan tombol apapun untuk mengakhiri")
    exit()

def clear_screen(): #Fungsi memperbarui CLI
    os.system('cls' if os.name == 'nt' else 'clear')

def log_parking_event(plate_number):
    global log_counter, log_entry
    log_counter += 1
    log_entry = { #Dictionary menampung log keluar & masuk kendaraan
        'Plat': plate_number,
        'Waktu Masuk': datetime.datetime.now()
    }
    log[log_counter] = log_entry

def log_exit_event(plate_number):
    global log_counter
    log_entry = log[log_counter]
    log_entry['Waktu Keluar'] = datetime.datetime.now()
    masuk = log_entry['Waktu Masuk']
    keluar = log_entry['Waktu Keluar']
    log_entry['Biaya'] = get_cost(masuk, keluar)

main()
```

C. Referensi

1. https://www.anandanesia.com/menghitung-waktu-dan-biaya-parkir-dengan-python/
2. https://srirahayuuu.blogspot.com/2016/10/python_11.html
3. https://www.erikadielsson.com/program-menghitung-biaya-parkir-python/