

Unguided dan Tugas Tipe 1
Modul 10 – RECURSIVE LIST

UGD

Atma Ride merupakan bisnis peminjaman motor di daerah Babarsari. Saat bisnis mulai berkembang lebih besar, pemilik Atma Ride membutuhkan sebuah program untuk pendataan peminjaman di system mereka. Program tersebut memiliki ketentuan umum sebagai berikut:

- Data harus disimpan didalam suatu recursive list dengan ketentuan tiap data tersebut harus memiliki **ID, Jenis Motor, Nama Peminjam, Tanggal Pinjam, Status, dan Harga Pinjam.**

Data
id: int
jenisMotor: String
namaPeminjam: String
tanggalPinjam: String
status: String
hargaPinjam: float

Beberapa menu yang diharuskan ada :

1. Input Data Peminjam

Di menu ini, user dapat menambahkan data ke dalam list peminjaman. Data yang diinputkan adalah **Jenis Motor(Matic/Manual), Nama Peminjam, dan Tanggal Pinjam.** Untuk **ID, Status, dan Harga Pinjam** akan otomatis terisi (**ID dimulai dari 1, 2, 3, dst untuk setiap penginputan data, Status secara default berisi dipinjam, dan Harga Rp 80.000 untuk matic dan Rp 75.000 untuk manual**). Menu ini juga terdiri dari 2 sub menu yaitu:

- **Input Baru →** akan menginputkan data baru dengan insertFirst atau insertLast
- **Input Setelah ID →** akan meminta ID, lalu menginputkan data baru setelah ID tersebut.
- **Menu Utama →** mengembalikan program ke menu utama.

2. Tampil Data

Menampilkan seluruh data untuk setiap list jika list peminjaman tidak kosong. Jika kosong akan menampilkan pesan error.

3. Cari Data

Menu ini akan meminta inputan **Nama Peminjam** untuk dicari dan ditampilkan datanya. Jika tidak ditemukan akan menampilkan error handling.

4. Pengembalian

Menu ini akan meminta inputan **Nama Peminjam** untuk dicari dan ditampilkan datanya. jika tidak ditemukan tampilkan error handling. Jika ditemukan kemudian user harus menginputkan jumlah uang sesuai harga dari data yang telah ditampilkan. Pemeriksaan jumlah uang memiliki beberapa ketentuan :

- Jika user menginputkan uang yang jumlahnya lebih kecil dari harga yang seharusnya, data harga saat ini akan dikurangi dengan jumlah uang hasil inputan user dan mengupdate status data menjadi Pending.
- Jika user menginputkan uang yang jumlahnya pas atau lebih besar dari harga yang seharusnya, data tersebut akan dihapuskan.

5. Exit

Program akan menampilkan nama, kelas, beserta npm pratikan dan kemudian mengakhiri program .

BONUS

1. **(10) Menambahkan menu Riwayat.** Menu ini bisa diakses apabila list riwayat tidak kosong. Jika list riwayat kosong akan menampilkan pesan error. Saat user melakukan pengembalian dan memasukkan jumlah uang yang lebih besar atau sama dengan harga di data tersebut, data akan dimasukkan ke dalam list riwayat dan menghapusnya dari list peminjaman. Menu ini juga dapat menampilkan total pendapatan yang didapat dari pembayaran dari user yang telah selesai maupun belum selesai.

Contoh :

Data saat ini

```
Data ke-1 | ID-1
Jenis Motor      : Matic
Nama Peminjam    : Anton
Tanggal Pinjam   : 08-10-2022
Status           : Dipinjam
Harga Pinjam     : Rp 80000.00

Data ke-2 | ID-2
Jenis Motor      : Manual
Nama Peminjam    : Andre
Tanggal Pinjam   : 17-04-2022
Status           : Dipinjam
Harga Pinjam     : Rp 75000.00
```

Data ke-1 belum lunas dan data ke-2 telah lunas

```
Data ke-1 | ID-1
Jenis Motor      : Matic
Nama Peminjam    : Anton
Tanggal Pinjam   : 08-10-2022
Status           : Pending
Harga Pinjam     : Rp 1000.00
```

Isi Riwayat

```
Total Pendapatan : 154000.00
```

```
Data ke-1 | ID-2
Jenis Motor      : Manual
Nama Peminjam    : Andre
Tanggal Pinjam   : 17-04-2022
Status           : Dipinjam
Harga Pinjam     : Rp 75000.00
```

2. (5) Menambahkan error handling pada setiap input tanggal dengan ketentuan :
 - Format tanggal : dd-mm-yyyy
 - Hari dan bulan masing-masing tidak dapat lebih besar dari 31 dan 12

Contoh format :

12-08-2022 (benar)

12/08/2022 (salah)

12-08-0001 (benar)

12-08-1 (salah)

3. (5) Melakukan modifikasi pada menu **Pengembalian** dimana menu ini akan meminta inputan tanggal pengembalian jika status data tersebut adalah dipinjam. Tanggal pengembalian tersebut memiliki deadline yaitu maksimal 2 hari setelah tanggal peminjaman. Jika tanggal pengembalian yang diinputkan user lewat dari 2 hari, denda akan berlaku yang dihitung setelah 2 hari deadline. Denda tersebut akan ditambahkan ke dalam harga peminjaman data tersebut dengan ketentuan setiap 1 hari yang terlewat akan dikenai denda Rp 40.000. Pada menu ini juga dilakukan pengecekan format saat user menginput tanggal pengembalian. Tanggal pengembalian juga tidak dapat diinputkan saat tanggal pengembalian lebih kecil dari tanggal peminjaman.

Contoh :

Data saat ini

```
Data ke-1 | ID-1
Jenis Motor      : Matic
Nama Peminjam    : Anton
Tanggal Pinjam   : 08-10-2022
Status           : Pending
Harga Pinjam     : Rp 10000.00
```

```
Data ke-2 | ID-2
Jenis Motor      : Manual
Nama Peminjam    : Andre
Tanggal Pinjam   : 17-04-2022
Status           : Dipinjam
Harga Pinjam     : Rp 75000.00
```

Melakukan pengembalian pada data dengan status Pending

```
Masukkan Nama Peminjam : Anton

ID-1
Jenis Motor      : Matic
Nama Peminjam    : Anton
Tanggal Pinjam   : 08-10-2022
Status           : Pending
Harga Pinjam     : Rp 10000.00

Masukkan Uang : 20000

Berhasil Dikembalikan - Kembalikan = Rp 10000.00
```

Error handling tanggal pengembalian dan mengembalikan lebih 1 hari dari deadline

```
Masukkan Nama Peminjam : Andre

ID-2
Jenis Motor      : Manual
Nama Peminjam    : Andre
Tanggal Pinjam   : 17-04-2022
Status           : Dipinjam
Harga Pinjam     : Rp 75000.00

Masukkan Tanggal Pengembalian : 01-03-2021

        Tanggal Pengembalian Tidak Sesuai [!]
Masukkan Tanggal Pengembalian : 20-04-2022

Anda Terkena Denda Sebanyak : Rp 40000.00
Total yang harus dibayar : Rp 115000.00

Masukkan Uang : 115000

Berhasil Dikembalikan - Kembalikan = Rp 0.00
```

KETENTUAN UGD

1. Harus memakai konsep recursive list bukan linked list 2 pada operasi – operasi umum di listnya.
2. Jika ditemukan operasi list yang seharusnya memakai recursive tetapi melakukan perulangan biasa maka akan dikurangi sebanyak 10 untuk setiap operasi.
3. Pastikan code dapat di compile tanpa adanya error

KETENTUAN BONUS

1. Menyelesaikan UGD + Semua Bonus selama pratikum: **UGD 120, TGS 110**
2. Tidak ada bonus laporan
3. Hanya diperbolehkan bertanya mengenai kejelasan soal bonus kepada asiste. Jika bertanya mengenai algoritma/cara pengerjaan maka bonus langsung hangus.
4. Bonus dapat dicek langsung selama di kelas dengan menghubungi asisten yang berjaga.

TUGAS

Selesaikan UGD + Bonus 1 + Bonus 2 + Bonus 3

LAPORAN

Menjelaskan algoritma program tugas maksimal 4 halaman tanpa cover. Wajib menyertakan nama, npm, dan kelas di atas dokumen. Jika tidak menyertakan identitas, nilai laporan -10.

Format Pengumpulan :

UGD : UGD10_Y_XXXXX

TGS : TGS10_Y_XXXXX

LAP : LAP10_Y_XXXXX

Bonus : UGD10_Y_XXXXX_Bonus1 / UGD10_Y_XXXXX_Bonus12 /
UGD10_Y_XXXXX_Bonus123

Y = Kelas, X = 5 digit NPM terakhir