**Laporan Tugas Proyek**

**Pemrograman Berorientasi Objek**

**Travel**



**Disusun Oleh:**

|  |  |
| --- | --- |
| 11322023 | Mananda Atalya Tambun |
| 11322028 | Christian Jhon Pranata Panjaitan |
| 11322033 | Eka Syahputra Lumbanraja |
| 11322058 | Dian Anggi Sitanggang |

**Institut Teknologi Del**

**Fakultas Vokasi**

**D3 Teknologi Informasi**

**Laguboti**

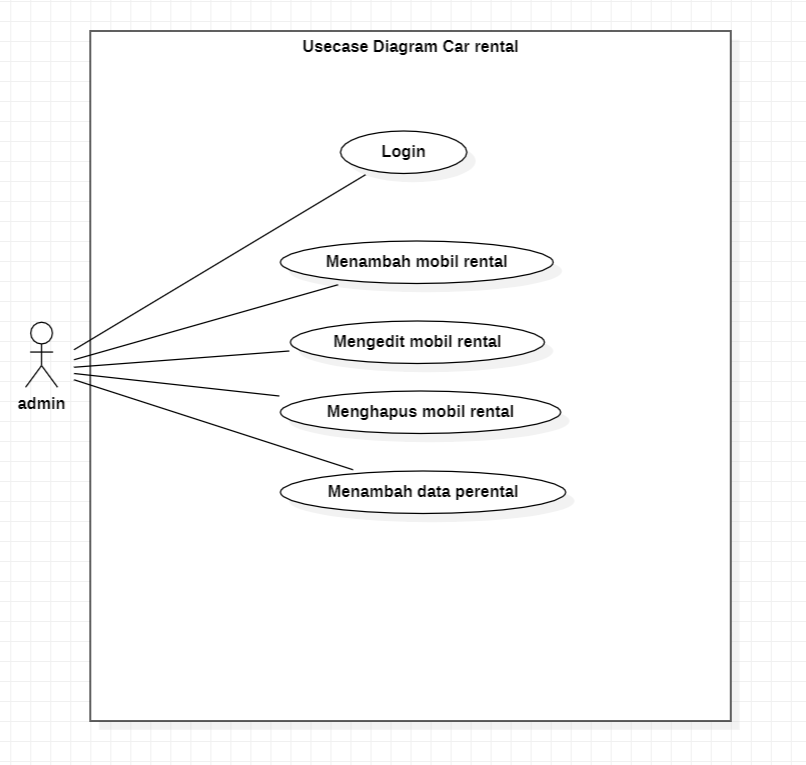
**2023**

**ABSTRAK**

Car rental adalah aplikasi berbasis desktop yang digunakan untuk melakukan peminjaman mobil yang akan dipinjam. Pada aplikasi ini hanya memiliki satu *role* yaitu admin. Admin akan menginput keseluruhan data mobil kedalam aplikasi, melakukan pengubahan data mobil serta penghapusan mobil yang tersedia didalam aplikasi. Untuk melakukan peminjaman terhadap mobil, admin akan menginput seluruhdata peminjam .Untuk aplikasi ini terdapat lima fungsi utama yang akan dijelaskan pada isi dokumen.

1. **FUNCTIONAL DESCRIPTION**

Pada bab ini menjelaskan deskripsi fungsional yang dilakukan aktor pada Aplikasi Car Rental. Pada bagian ini juga digambarkan usecase diagram *travel.* *Use case* diagram menggambarkan hal apa saja yang dapat dilakukan oleh aktor terhadap sistem, dapat dilihat pada gambar 1 berikut.



Gambar 1. Use Diagram Aplikasi Car rental

Fungsi – fungsi yang ada pada aplikasi *Car rental* yaitu:

1. Fungsi Login.

Fungsi ini digunakan oleh *admin* untuk masuk kedalam sistem dan dapat mengakses data didalam sistem.

1. Fungsi Menambah Mobil rental

Fungsi ini digunakan oleh *admin* untuk menambah data mobil rental kedalam sistem

1. Fungsi Mengedit Mobil rental.

Fungsi ini digunakan oleh *admin* untuk mengedit data mobil rental yang akan diubah.

1. Fungsi Menghapus Data Mobil rental.

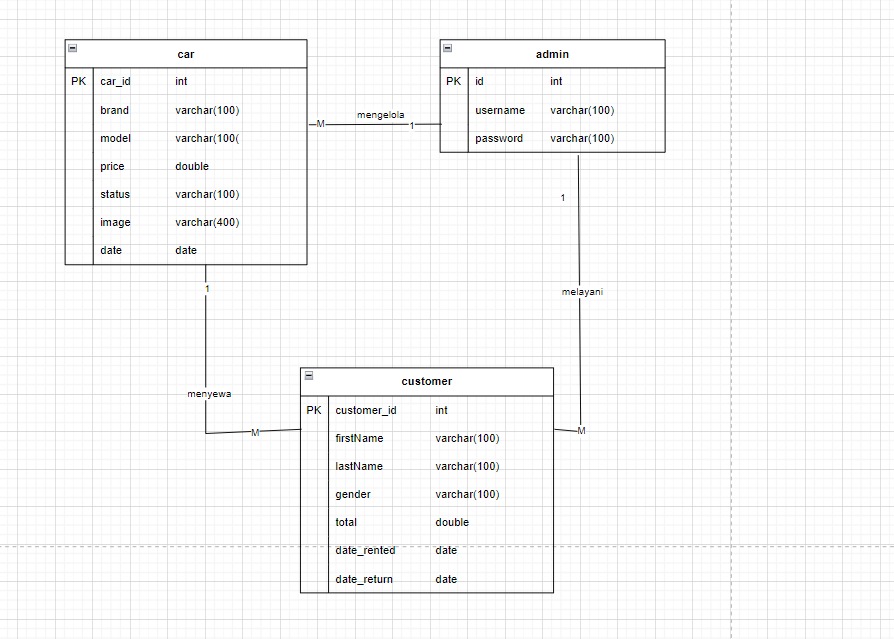
Fungsi ini digunakan oleh *admin* untuk menghapus data peminjaman mobil yang sudah diinput oleh admin.

1. Fungsi Menambah Data Perental Mobil.

Fungsi ini digunakan oleh *admin* untuk mengisi data peminjam/perental mobil beserta mobil yang dipinjam.

1. **DESIGN CONCEPTUAL DATA MODEL**

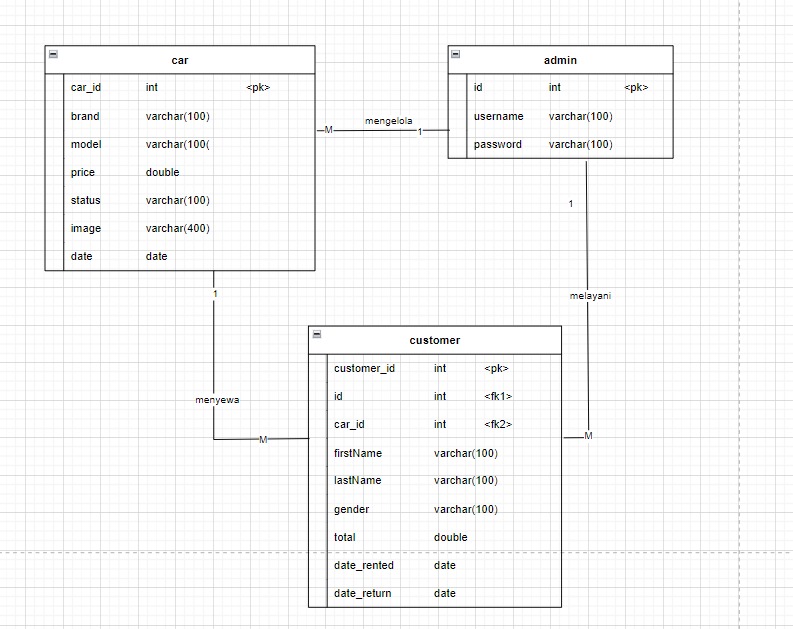
*Conceptual data model* berguna untuk memodelkan struktur logis dari keseluruhan aplikasi data. *Conceptual data model* dari Aplikasi *Car rental* terlampir pada gambar 2.



Gambar 2. Conceptual Data Model Aplikasi Car rental

1. **DESIGN PHYSICAL DATA MODEL**

*Physical Data Model (PDM)* berguna untuk menggambarkan hubungan antar tabel secara fisikal. *Physical data model* dari Aplikasi *Car rental* terlampir pada gambar 3.



Gambar 3. Physical Data Model Aplikasi Car rental

1. **DESIGN TABLE STRUCTURE**

Bab ini menjelaskan keseluruhan tabel yang telah dirancang untuk aplikasi Travel. Setiap tabel akan berisi nama tabel, deskripsi isi, type dan *length*, dan *primary key.*

## **4.1 Tabel Car**

Identifikasi/Nama : Car

Deskripsi Isi : Tabel ini berisi atribut mobil yang akan dirental.

Jenis : Tabel master

Volume : 8 *field*

Primary Key : id

Data *dictionary* tabel *book* terlampir dalam tabel 1.

Table 1. Tabel Structure Book

| ***Field*** | **Deskripsi** | ***Type and Length*** | **Boleh NULL** | ***Default*** | **Keterangan** |
| --- | --- | --- | --- | --- | --- |
| car\_id | Berisi id | INT | *No* | None | *Primary key* |
| brand | Berisi brand mobil | VARCHAR | *No* | None | *attribute non key* |
| model | Berisi model mobil | VARCHAR | *No* | None | *attribute non key* |
| price | Berisi harga mobil | Double | *No* | None | *attribute non key* |
| status | Berisi status mobil | VARCHAR | *No* | None | *attribute non key* |
| date | Berisi tanggal transaksi mobil | DATE | *Yes* | NULL | *attribute non key* |

## **4.2 Tabel Admin**

Identifikasi/Nama : Admin

Deskripsi Isi : Tabel ini berisi tentang admin

Jenis : Tabel master

Volume : 3 *field*

Primary Key : id

Data *Dictionary* tabel admin terlampir dalam tabel 2.

Table 2. Tabel Structure admin

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| ***Field*** | **Deskripsi** | ***Type and Length*** | **Boleh NULL** | ***Default*** | **Keterangan** |
| id | Berisi id | INT | *No* | None | *Primary key* |
| username | Berisi username pengguna | VARCHAR | *No* | None | *attribute non key* |
| password | Berisi password pengguna | VARCHAR | *No* | None | *attribute non key* |

## **4.3 Tabel Customer**

Identifikasi/Nama : Customer

Deskripsi Isi : Tabel ini berisi tentang customer / pembeli

Jenis : Tabel master

Volume : 10 *field*

*Primary Key* : customer\_id

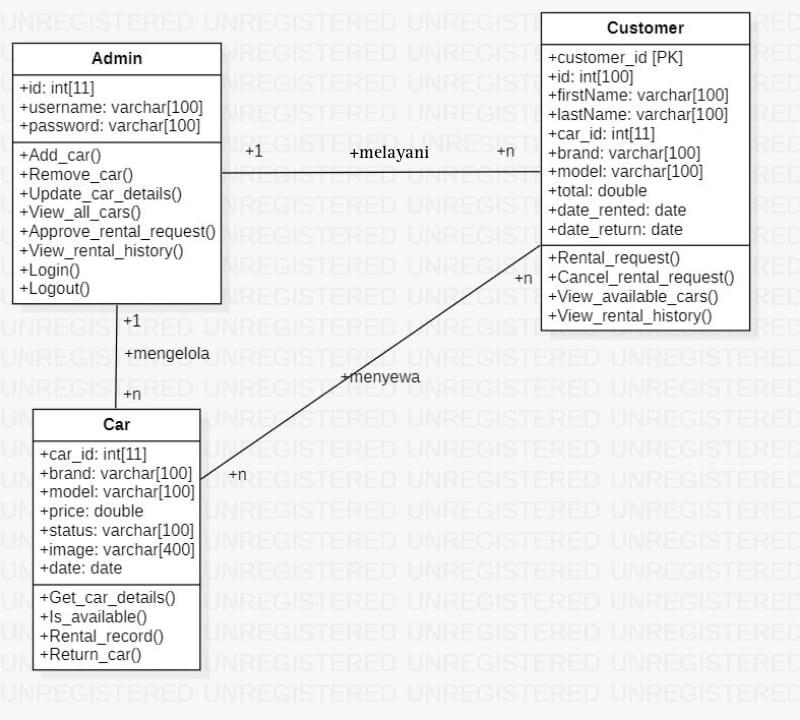
Data *Dictionary* tabel take terlampir dalam tabel 3.

Table 3. Tabel Structure take

| ***Field*** | **Deskripsi** | ***Type and Length*** | **Boleh NULL** | ***Default*** | **Keterangan** |
| --- | --- | --- | --- | --- | --- |
| customer\_id | Berisi  id customer | INT | *No* | None | *primary key* |
| firstName | Berisi nama pertama peminjam | VARCHAR | *Yes* | None | *attribute non key* |
| lastName | Berisi nama terakhir peminjam | VARCHAR | *Yes* | None | *attribute non key* |
| gender | Berisi jenis kelamin peminjam | VARCHAR | *Yes* | None | *attribute non key* |
| car\_id | Berisi id car yang dipinjam | INT | *No* | None | *foreign key* |
| brand | Berisi brand mobil | VARCHAR | *Yes* | None | *attribute non key* |
| model | Berisi model car | VARCHAR | *Yes* | None | *attribute non key* |
| total | Berisi jumlah total item car | DOUBLE | *Yes* | None | *attribute non key* |
| date\_rented | Berisi tanggal peminjaman | DATE | *Yes* | NULL | *attribute non key* |
| date\_return | Berisi tanggal pengembalian | DATE | *Yes* | NULL | *attribute non key* |

1. **DESIGN CLASS DIAGRAM**

*Class diagram* adalah diagram yang menggambarkan suatu model data untuk suatu *website*, tidak peduli apakah model data tersebut sederhana maupun kompleks. *Class diagram* Aplikasi *Car rental* dapat dilihat pada gambar 4 berikut.



Gambar 4. Class Diagram Aplikasi Car rental

Dari pembentukan *class diagram* ini, terdapat 3 *class* dengan masing-masing attribute dan method yaitu:

1. Admin memiliki atribut: id, username , dan password . Dengan methodnya  
   yaitu login,logout,Add\_car , Remove\_car , Update\_car\_details, View\_all\_cars , approve\_rental\_request ,view\_rental\_history.
2. Car memiliki atribut: car\_id, brand, model, price, status, image, dan Date. Dengan methodnya yaitu Get\_car\_details, is\_available ,rental\_record, return\_car.
3. Customer memiliki atribut: customer\_id, , firstName, lastName, car\_id, brand,model,total,date\_rented,date\_return .Dengan methodnya yaitu rental\_request ,cancel\_rental\_request , view\_available\_car , view\_rental\_history.
4. **OOP IMPLEMENTATION**
   1. **Abstraction**

Untuk OOP implementation pada aplikasi Car rental terdapat konsep Abstraction dimana dengan menggunakan abstraction kita dapat memecah kompleksitas pada sebuah program menjadi unit yang terkecil sehingga kita akan mudah untuk dimaintenance. Penerapan Abstraction dapat dilihat pada gambar 5

Gambar 5. Gambar Penerapan Abstraction



* 1. **Encapsulation**

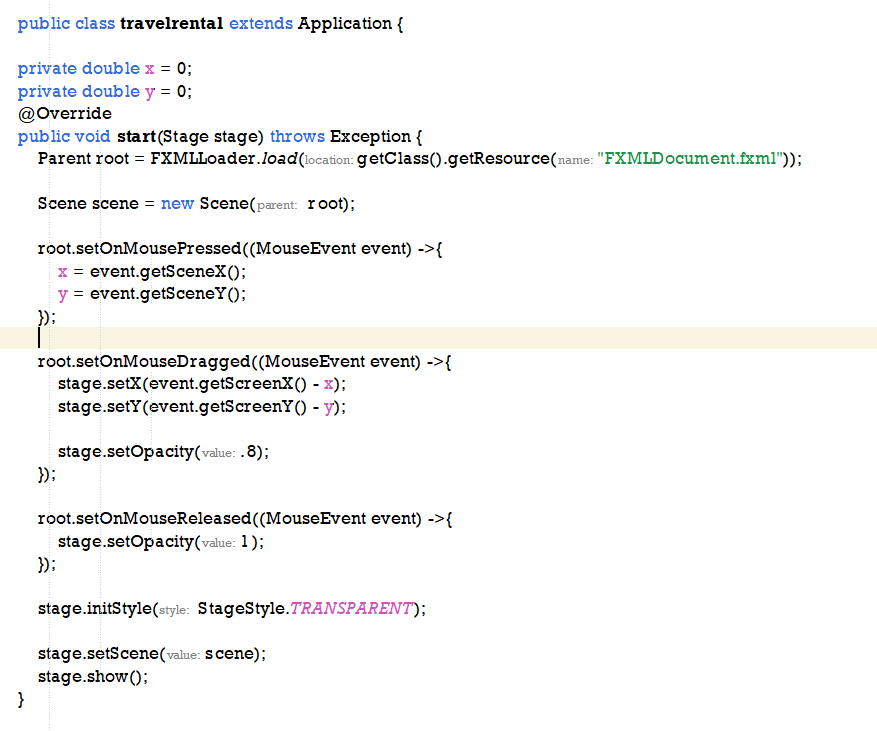
Pada contoh gambar nomor 6. yang diberikan, terdapat dua variabel amount dan balance yang bersifat privat (private). Penggunaan access modifier private tersebut adalah bagian dari konsep encapsulation di dalam pemrograman berorientasi objek (OOP).

Encapsulation adalah salah satu prinsip OOP yang mengizinkan penyembunyian rincian implementasi dari luar kelas dan hanya mengekspos fungsionalitas yang diperlukan. Dengan menggunakan private pada variabel amount dan balance, kita membatasi akses langsung ke variabel tersebut dari luar kelas. Oleh karena itu, variabel-variabel tersebut hanya dapat diakses dan dimodifikasi oleh metode-metode yang ada di dalam kelas yang sama.

Gambar 6. Gambar Penerapan Encapsulation

* 1. **Inheritance**

Untuk OOP implementation pada aplikasi Car rental terdapat konsep Inheritance dimana terdapat keyword “extends” yang menunjukkan bahwa kita membuat kelas baru yang berasal dari kelas yang sudah ada. Jadi kelas travelrental yang menggunakan extends Application akan menggunakan metode Application() untuk membukanya. Penerapan Inheritance dapat dilihat pada gambar 7.



* 1. **Polymorphism**

Untuk OOP implementation pada aplikasi Car Rental terdapat konsep Polymorphism yang merupakan sebuah konsep OOP dimana class memiliki banyak “bentuk” method yang berbeda meskipun nama yang sama. Penerapan Polymorphism dapat dilihat pada gambar 8.

****

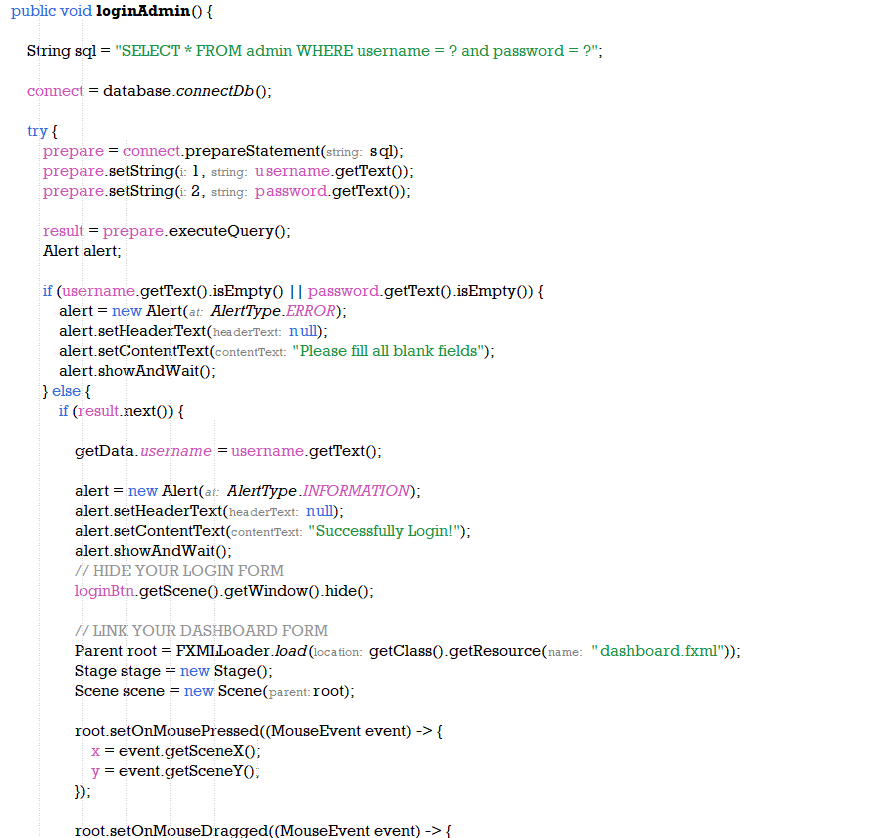
* 1. **Interface**

Untuk OOP implementation pada aplikasi Car Rental terdapat konsep Interface, dimana terdapat keyword “implements” yang menunjukkan bahwa class dimana semua method didalamnya diimplementasikan ulang oleh class yang mengimplementasikanya. Penerapan dapat dilihat pada gambar 9.



* 1. **Exception handling**

Untuk OOP implementation pada aplikasi Car rental terdapat konsep Exception Handling yaitu terdapat keyword "try" yang digunakan untuk mencoba menjalankan block program kemudian mengenai dimana munculnya kesalahan yang ingin diproses dan terdapat juga keyword "catch" yang digunakan untuk menangkap kesalahan atau bug yang terjadi dalam block try. Penerapan Exception

****

****

1. **Implementasi keseluruhan dari aplikasi manajemen persewaan mobil dengan antarmuka pengguna berbasis JavaFX.**

1. DashboardController:

Inisialisasi dan Tampilan: initialize method digunakan untuk menginisialisasi komponen antarmuka pengguna, termasuk mengatur grafik, tabel, dan memanggil fungsi untuk menampilkan data.

Navigasi: switchForm method digunakan untuk beralih antara tampilan beranda, daftar mobil yang tersedia, dan formulir sewa mobil.

Fungsi Beranda: Terdapat beberapa fungsi seperti homeAvailableCars, homeTotalIncome, homeTotalCustomers, homeIncomeChart, dan homeCustomerChart yang mengambil data dari database dan menampilkan informasi di beranda.

2. Fungsi Manajemen Mobil Tersedia (availableCar):

Import Gambar dan Penyimpanan Data: availableCarImportImage digunakan untuk mengimpor gambar mobil dan menampilkannya. availableCarAdd menambahkan data mobil ke database, termasuk meng-handle gambar.

Update dan Hapus Data: availableCarUpdate dan availableCarDelete digunakan untuk memperbarui dan menghapus data mobil dari database.

Pencarian dan Penyaringan: availableCarSearch digunakan untuk mencari data mobil berdasarkan kriteria tertentu.

Tampilan dan Seleksi Data: availableCarShowListData menampilkan data mobil di tabel, dan availableCarSelect memungkinkan pengguna untuk memilih mobil dari tabel.

3. Fungsi Manajemen Penyewaan (rent):

Penyewaan dan Pembayaran: rentPay digunakan untuk menyimpan informasi penyewaan ke database dan memperbarui status mobil menjadi "Not Available". rentAmount digunakan untuk menghitung jumlah pembayaran berdasarkan harga dan durasi penyewaan.

Inisialisasi dan Tampilan: rentCarShowListData menampilkan data mobil yang dapat disewa di tabel. rentCarCarId, rentCarBrand, dan rentCarModel digunakan untuk mengisi combo box dengan data mobil yang tersedia.

Clear dan Validasi Formulir: rentClear digunakan untuk membersihkan formulir sewa mobil. rentCustomerId menghasilkan ID pelanggan baru untuk setiap penyewaan.

4. Fungsi Umum:

Inisialisasi User dan Tampilan: displayUsername menampilkan nama pengguna pada antarmuka.

Logout: logout memungkinkan pengguna untuk keluar dari aplikasi.

Minimize dan Close Window: minimize dan close digunakan untuk meminimalkan dan menutup jendela aplikasi.

Implementasi keseluruhan ini membentuk aplikasi yang dapat digunakan untuk manajemen persewaan mobil dengan fitur-fitur seperti penyimpanan data mobil, penyewaan, dan tampilan statistik pendapatan. Aplikasi ini menggunakan JavaFX untuk antarmuka pengguna dan mengakses database untuk menyimpan dan mengambil data.

1. **LINK YOUTUBE**

[**https://youtu.be/-v\_gOY6wh\_o**](https://youtu.be/-v_gOY6wh_o)