**Project Name :**

**Python Project Pacmann - Super Cashier  
  
Oleh : Yendri Kilauan Purnama  
Student : #pacmann-peduli-awardee (Thank You Pacmann)  
Batch 1.**

**Background Project**

Andi adalah seorang pemilik supermarket besar di salah satu kota di Indonesia. Andi memiliki rencana untuk melakukan perbaikan proses bisnis, yaitu Andi akan membuat sistem kasir yang self-service di supermarket miliknya. Sehingga customer bisa langsung memasukkan item yang dibeli, jumlah item yang dibeli, dan harga item yang dibeli dan fitur yang lain. Sehingga customer yang tidak berada di kota tersebut bisa membeli barang dari supermarket tersebut. Setelah Andi melakukan riset, ternyata Andi memiliki masalah, yaitu Andi membutuhkan Programmer untuk membuatkan fitur - fitur agar bisa sistem kasir self-service di supermarket itu bisa berjalan dengan lancar.

**Deskripsi Task**

Akhirnya Andi meminta tolong kepada teman-teman selaku programmer Python untuk membuat program yang menyelesaikan problem tersebut.

Jika ada yang berbelanja, begini journey customer dalam membantu orang yang berbelanja tersebut.

**1. Customer membuat ID transaksi customer berikut:**

* Dengan membuat objek dari class transct\_123 = Transaction()

**2. Kemudian customer memasukkan nama item, jumlah item, dan harga barang**

* Masukkan item yang ingin dibeli.

add\_item([<nama item>, <jumlah item>, <harga per item>])

**3. Jika terjadi kesalahan dalam memasukkan nama item atau jumlah item atau harga item tetapi tidak ingin menghapus itemnya, Customer bisa melakukan:**

a. Update nama item dengan method:

update\_item\_name(<nama item>, <update nama item>)

b. Update jumlah item dengan method:

update\_item\_qty(<nama item>, <update jumlah item>)

c. Update harga item dengan method:

update\_item\_price(<nama item>, <update harga item>)

**4. Jika batal membeli item belanja, customer bisa melakukan:**

a. menghapus salah satu item dari nama item dengan method: delet\_item(<nama\_item>)

| **Item** | **jumlah item** | **harga per item** | **harga total** |
| --- | --- | --- | --- |
| mobil | 2 | 100000 | 200000 |
| (item yang dihapus) |  |  |  |
| tempe | 3 | 3000 | 9000 |

Ketika menghapus salah satu nama item, maka jumlah item dan harga per item pada baris/list tersebut akan ikut terhapus

b. Langsung menghapus semua transaksi atau reset transaksi dengan method:

reset\_transaction()

**5. Customer sudah selesai dengan berbelanja online nya, tetapi Customer masih ragu apakah harga barang dan nama barang yang diinput sudah benar. Bisa saja customer melakukan kesalahan dalam melakukan input, semisal sudah melakukan input harga barang tetapi lupa untuk input nama barangnya. Bisa menggunakan method:**

check\_order(). Dengan ketentuan:

a. Akan mengeluarkan pesan "Pemesanan Sudah Benar"

b. Akan mengeluarkan pesan "Terdapat kesalahan input data"

c. Keluarkan output berikut:

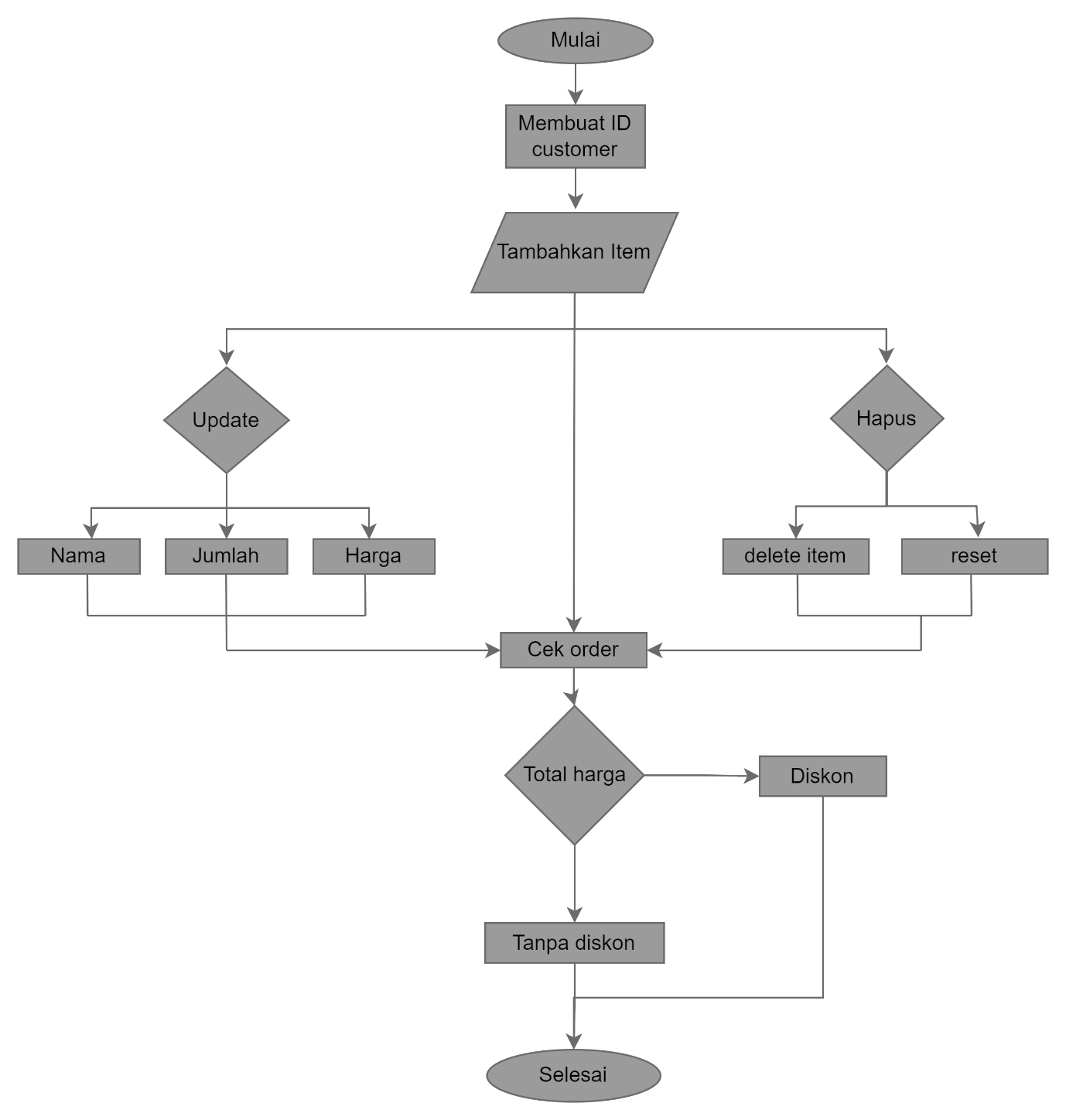
| **Item** | **jumlah item** | **harga per item** | **harga total** |
| --- | --- | --- | --- |
| mobil | 2 | 100000 | 200000 |
| mie | 1 | 5000 | 5000 |
| tempe | 3 | 3000 | 9000 |

**6. Setelah melakukan pengecekan, customer bisa menghiting total belanja yang sudah dibeli menggunakan method total\_price() dengan ketentuan:**

* Jika total belanja diatas 200.000 akan mendapat diskon 5%
* Jika total belanja diatas 300.000 akan mendapat diskon 8%
* Jika total belanja diatas 500.000 akan mendapat diskon 10%

Andi juga memberikan pesan kepada teman-teman kalau diberi kebebasan untuk menambahkan fitur yang lain apabila masih terdapat fitur yang belum tercover dalam sistem tersebut.

# Flowchart

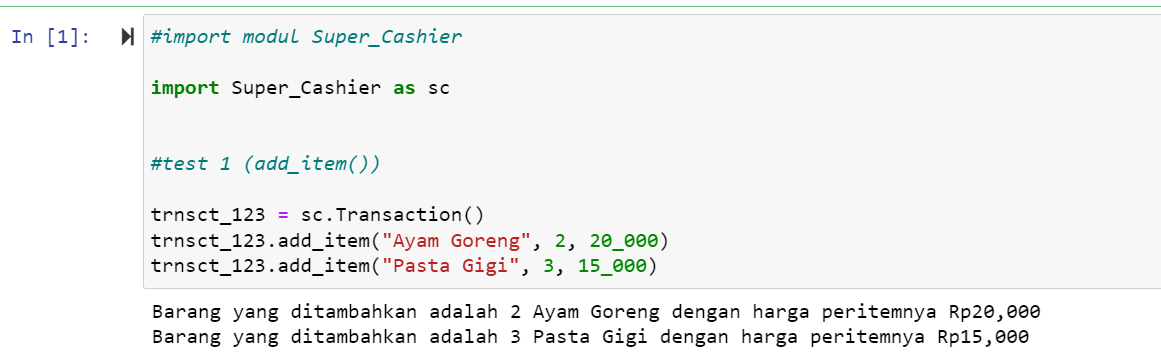


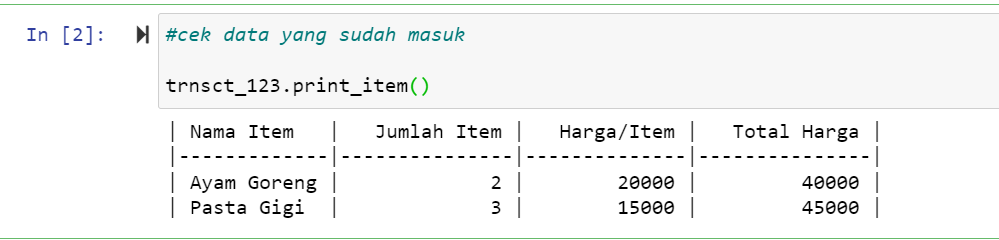
**Code (penjelasan tentang fungsinya)**

* class Transaction: Kelas digunakan untuk menampung method-method yang berisi feature requirements yang dibutuhkan.
* def \_\_init\_\_(self): fungsi untuk menginisialisasi sebuah atribut data pada class Transaction. Pada fungsi ini terdapat satu parameter self.data\_transaksi (dict) yang merupakan variabel untuk menampung data transaksi customer yang masuk.
* def add\_item(self, nama, jumlah, harga): Fungsi untuk memasukkan item yang ingin dibeli. Dengan input nama, jumlah, dan harga peritem.
* print\_item(self): Fungsi untuk mencetak item-item yang masuk dengan memberikan input berupa tabulate.
* update\_item\_name(self, nama, nama\_baru): Fungsi untuk mengupdate nama item.
* update\_item\_qyt(self, nama, jumlah\_baru): Fungsi untuk mengupdate jumlah item.
* def update\_item\_price(self, nama, harga\_baru): Fungsi untuk mengupdate harga per item .
* def delete\_item(self, nama): Fungsi untuk menghapus salah satu item .
* def reset\_transaction(self): Fungsi untuk mereset atau menghapus semua data transaksi yang ada.
* def check\_order(self):Fungsi untuk mengecek apakah data transaksi yang diinput sudah benar atau belum. Pengecekan dilakukan dengan mengecek apakah nama yang diinput bukan string kosong " " dan jumlah serta harga per item harus dalam tipe data integer.
* def total\_price(self): Fungsi untuk menghitung total belanja yang harus dibayar. Dengan memberikan diskon dengan ketentuan:
  + Jika totalbelanja di atas Rp200.000 maka akan mendapatkan diskon 5%.
  + Jika totalbelanja di atas Rp300.000 maka akan mendapatkan diskon 8%.
  + Jika totalbelanja di atas Rp500.000 maka akan mendapatkan diskon 10%.

# Test Code

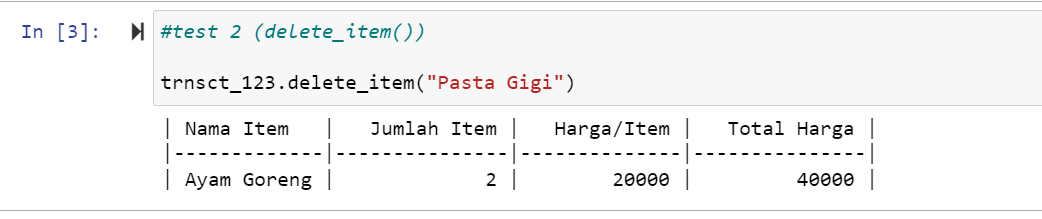
#### Test 1:

Customer ingin menambahkan dua item dengan method add\_item() kemudian mencetak list data yang sudah masuk dengan print\_item()  
  




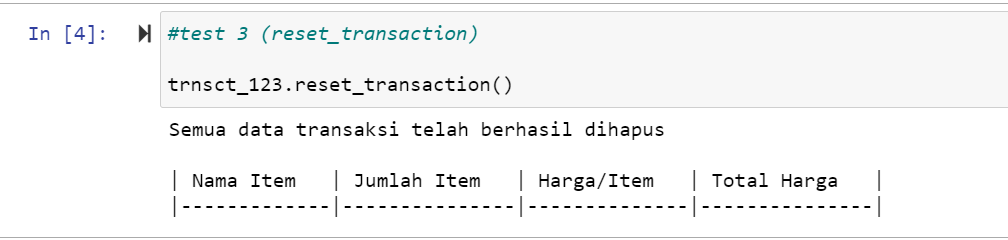
#### Test 2:

Customer salah membeli barang, sehingga ingin menghapus item "Pasta Gigi" dengan method delete\_item()



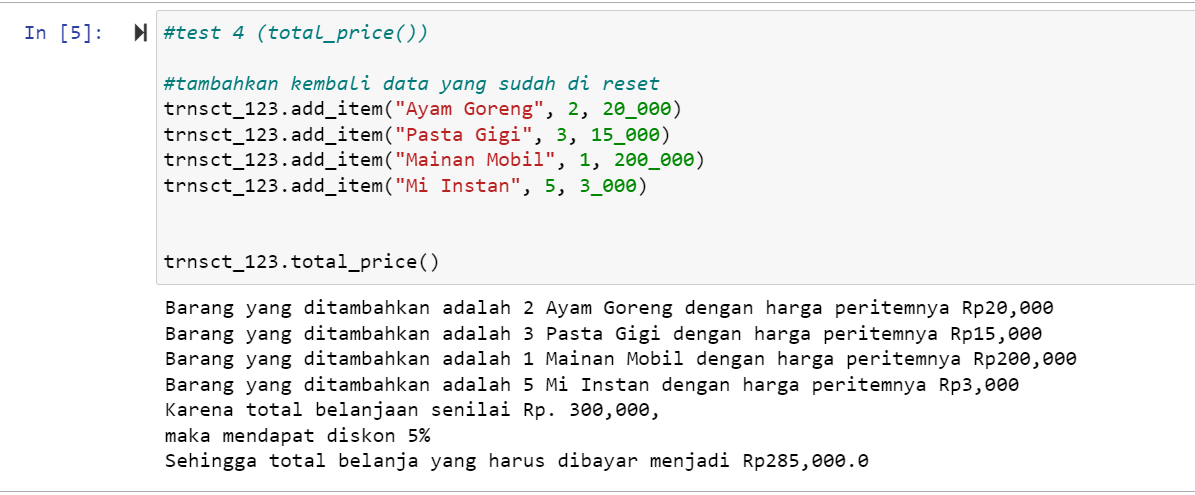
#### Test 3:

Ternyata customer salah memasukkan semua item yang ingin dibelanjakan, sehingga customer ingin menghapus semua data transaksi yang masuk dengan method reset\_transaction()



#### Test 4:

Customer menginput kembali item-item yang ingin dibeli, kemudian akan menghitung semua total belanja yang harus dibayar dengan method total\_price()



## **Conclusion**

Sistem kasir self-service ini dibuat untuk mengelola proses transaksi di supermarket agar lebih mudah, efektif dan efisien. Dalam project ini telah dibuat sitem kasir yang dengan beberapa fitur dapat membantu proses transaksi jarak jauh diantaranya dapat menginput item yang ingin dibeli, memperbarui item, menghapus item, membatalkan transaksi dan mengecheck total harga keseluruhan di daftar belanja. Sehingga hal ini membuat customer dapat dengan mudah belanja di supermarket Andi tanpa harus datang langsung ke supermarket.