**PROGRAM RestaurantCashRegister Simulation**{ Deskripsi : Program simulasi cara kerja Mesin Kasir Restoran}

**KAMUS**{df : array of integer and string}  
{user, pembeli : string}  
{pinbaru, PIN, uang, result : integer}

**ALGORITMA**

{ Kumpulan fungsi dan prosedur }

**function** menu**() #int** {Mencetak menu ke layer dan menerima input pilihan dari user}  
 **KAMUS LOKAL** { menu : integer}  
 **ALGORITMA** output (”””=====================  
 ╔═══╗╔═══╗ ╔╗  
 ║╔═╗║║╔═╗║ ╔╝╚╗  
 ║╚═╝║║╚═╝╠══╦═╩╗╔╬══╗  
 ╚══╗║║╔╗╔╣║═╣══╣║║╔╗║  
 ╔══╝║║║║╚╣║═╬══║╚╣╚╝║  
 ╚═══╝╚╝╚═╩══╩══╩═╩══╝  
 =====================  
 Pilih menu:  
 1. Pembelian  
 2. Ganti PIN  
 3. Tambah Admin”””)  
 input (menu)  
 **return** menu

**function** prompt **() #bool**  
 { Memberi pilihan kepada user apakah akan melanjutkan program atau tidak }  
 **KAMUS LOKAL** { ask : string }  
 **ALGORITMA** input (ask)  
 **if** (ask == ‘y’) **then**  
 **return** 1 **else  
 output** (“Terima kasih telah menggunakan Mesin Kasir ini.”)  
 **return** 0  
 **endif**

**function** formatrupiah(nilai)  
 { Mengubah penulisan rupiah sesuai format yang benar }  
 y = nilai  
 **if** (len(y) <= 3) **then**  
 **return** “Rp” + y  
 **else**  
 p = y[-3:]  
 q = y[:-3]  
 **return** formatrupiah(q) + “.” + p  
 **endif**

**procedure** gantiPIN(user, pinbaru)  
 { Mengganti pin user untuk proses LOGIN berikutnya }  
 **KAMUS LOKAL** { user, pinbaru : string }  
 **ALGORITMA** **if** (len(pinbaru == 6) and (pinbaru.isnumeric()) **then** df[user][0] = pinbaru  
 print(“Berhasil mengubah PIN untuk {user}”)  
 writefile(“data.csv”)  
 **else** output(“PIN baru yang anda masukkan tidak ”)  
 **endif**

**procedure** tambahadmin(useradm, pinadm)  
 { Menambah data user pada database beserta PIN-nya }  
 **KAMUS LOKAL** { user, pinbaru : string }  
 **ALGORITMA** headerCSV = [“username”, “PIN”]  
 data = [“username” : useradm, “PIN” : pinadm]  
 **if** (len(pinadm) == 6 and pinadm.isnumeric)**then**  
 **with** open(“data.csv”) as f\_object  
 dictwriter\_object = DictWriter(f\_object, fieldname=headerCSV)  
 dictwriter\_object.writerow(data)  
 f\_object.close()  
 output({useradm} telah ditambahkan sebagai Admin.)  
 **else**  
 output(“PIN admin baru yang anda masukkan tidak valid. Coba lagi nanti.”)  
 **endif**

**procedure** userblock(user)  
 { Menghapus data pada user pada database }  
 **KAMUS LOKAL** { user : string }  
 **ALGORITMA** df = readfile(“data.csv”)  
 df = df[df[“username”] != user]  
 writefile(“data.csv”)

**procedure** sendto(date, time, pembeli, user, result)  
 { Menambah database pembelian berdasarkan pembelian yang telah berlangsung }  
 **KAMUS LOKAL** { date, time, pembeli, user : string, pemasukan : integer }  
 **ALGORITMA** headerCSV = ['date', 'time', 'nama', 'staff', 'pemasukan']  
 data\_pembelian = {'date': date, 'time' : time, 'nama' : pembeli, 'staff' : user, 'pemasukan' : result }  
 **with** open('data\_pembelian.csv', 'a', newline='') as f\_object:   
 dictwriter\_object = DictWriter(f\_object, fieldnames=headerCSV)  
 dictwriter\_object.writerow(data\_pembelian)  
 f\_object.close()

**procedure** ceksaldo()  
 { Mencetak seluruh pemasukan }  
 **KAMUS LOKAL** { pemasukan : integer }  
 **ALGORITMA** df = pd.read\_csv(“data\_pembelian.csv”)  
 saldo = df[“pemasukkan”].sum()  
 output [“Total pemasukan 9 Resto adalah Rp{formatrupiah(saldo)},00”]

{ inisialisasi dataframe }  
df = readfile(“data.csv”)

{ Program utama }

{ Penyocokan user dengan yang ada di database }  
**while** (user not in df[“PIN”]) **do** input (user)  
 **if** (rek not in df[“PIN”]) **then**  
 output(“Username anda tidak valid. Silahkan coba lagi.”)  
 **endif  
endwhile**

{ Autentikasi PIN }  
**while** tries < 3 **do  
 try** input(pin)  
 **if** (pin == df[user][0]) **then  
 break**  
 **else** {pin == df[user][0]}  
 raise **KeyError** **endif  
 except** KeyError  
 **if** (tries < 2) **then** output(“PIN anda salah. Coba lagi. (" + str(2-tries) + " percobaan lagi).”)  
 **else** output(“User {user} terblokir. Silahkan hubungi admin yang sah.”)  
 userblock(user)  
 **exit()  
 endif** tries = tries + 1  
**endwhile**

{ Main Utama }  
me = 0  
**while** true **do** me == menu()  
  
 **if** (me == 1) **then** { Pilihan menu pertama, program struk belanja }  
 **class** MenuItem  
 **procedure** \_\_init\_\_(self, name, price)  
 self.name = name  
 self.price = price  
 **procedure** get\_total\_price(self, count)  
 total\_price = self.price \* count  
 **if** count> 3:  
 total\_price \*= 0.9  
 **endif**  
 **return** round(total\_price)  
 **class** Food(MenuItem)  
 **procedure** \_\_init\_\_(self, name, price, calorie\_count)  
 **procedure** info(self)  
 return self.name + ': ' + formatrupiah(self.price) + ',00' + ' (' + self.calorie\_count + 'kkal)'  
 **class** Drink(MenuItem)  
 **procedure** \_\_init\_\_(self, name, price, volume)  
 **procedure** info(self)  
 return self.name + ': ' + formatrupiah(self.price) + ',00' + ' (' + self.volume + 'mL)'  
 { Membuat instance untuk class Food }  
 food1 = Food('Seblak', 5000, 330)  
 food2 = Food('Batagor', 4000, 450)  
 …  
 foods = [food1, food2, …]  
 { Membuat instance untuk class Drink }  
 drink1 = Drink('Kopi', 3000, 180)  
 drink2 = Drink('Jus Jeruk', 4000, 350)  
 …  
 drinks = [drink1, drink2, …]  
 output(“MAKANAN”)  
 index = 0  
 **for** food **in** foods  
 output(index + “. ” + food.info())  
 index += 1  
 **endfor**  
 output(“MINUMAN”)  
 index = 0  
 **for** drink **in** drinks  
 output(index + “. ” + drink.info())  
 index += 1  
 **endfor**  
 input(pembeli)  
 input(food\_order)  
 **while** food\_order < 0 or food\_order >= len(foods)  
 output(“Makanan yang anda pilih tidak berada pada menu.”)  
 input(food\_order)  
 **endwhile**  
 selected\_food = foods[food\_order]  
 input(drink\_order)  
 **while** drink\_order < 0 or drink\_order >= len(drinks)  
 output(“Minuman yang anda pilih tidak berada pada menu.”)  
 input(drink\_order)  
 **endwhile**  
 selected\_drink = drinks[drink\_order]   
 output(“Anda akan membeli {selected\_food.name} dan {selected.drink.name}”)  
 input(count)  
 isDiskon = “Tidak”  
 **if** count > 3  
 isDiskon = “Ya”  
 **endif**  
 result = selected\_food.get\_total\_price(count) + selected\_drink.get\_total\_price(count)  
 output(“Harga total adalah {formatrupiah(result)}”)  
 input(uang)  
 **if** uang >= result  
 define date, time  
 output(””””Output :  
=================================   
------ S T R U K B E L I ------  
-------- (9 R E S T O) --------  
=================================  
Date : {date}  
Time : {time}  
Staff : {user}  
=================================  
Nama : {pembeli}  
Makanan : {selected\_food.name}  
Minuman : {selected\_drink.name}  
Jumlah paket : {count}  
=================================  
Diskon 10% : {isDiskon}  
=================================  
Total : {formatrupiah(result) + ',00'}  
Bayar : {formatrupiah(uang) + ',00'}  
=================================  
Kembali : {fm.formatrupiah(uang-result) + ',00'}  
=================================  
--- T E R I M A K A S I H ---  
=================================

”””)

sendto(date, time, pembeli, user, result)  
 **else**  
 output(“Uang yang anda miliki tidak cukup. Coba lagi nanti.”)  
 **endif**  
 **if** (prompt())==0) **then**  
 **break**  
 **endif  
  
 else if** (me == 2) **then** { Pilihan menu kedua, ganti PIN dari user }  
input(pinbaru)  
 gantiPIN(user, pinadm)  
 **if** (prompt())==0) **then**  
 **break**  
 **endif  
  
 else if** (me == 3) **then** { Pilihan menu ketiga, menambah admin kasir }input(useradm)  
 input(pinadm)  
 tambahadmin(useradm, pinadm)  
 **if** (prompt())==0) **then**  
 **break**  
 **endif  
  
 else if** (me == 4) **then** { Pilihan menu keempat, mencetak total pemasukan }  
 ceksaldo()  
 **if** (prompt())==0) **then**  
 **break**  
 **endif  
  
 else** { Bila masukan nomor menu tidak valid }  
 output(“Nomor menu tidak valid”)  
 **if** (prompt()) == 0) **then  
 break  
 endif  
 endif  
endwhile**