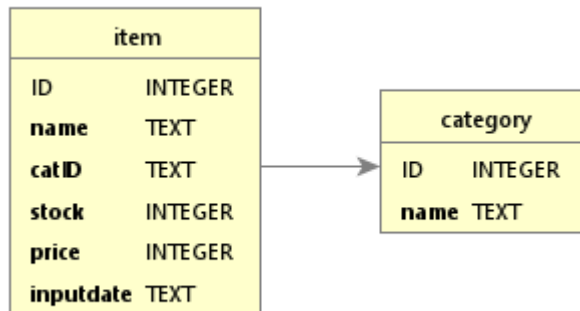


5210411383\_ALLAN BIL FAQIH

ALPROPRAK | Dokumentasi Program Responsi 2

## Entity Relationship Diagram



## Penjelasan Fungsi

1. Fungsi Utama | def sumberjaya()

### Codes

```
10 def sumberjaya():
11     main_menu = ["[1] MASUK KE MENU ITEM",
12                 "[2] MASUK KE MENU KATEGORI",
13                 "[3] KELUAR DARI APP"]
14
15 > def squek(): ...
49
50 > def showCat(): ...
59
60 > def item(): ...
271
272 > def cat(): ...
407
408 > def showMenu(): ...
411
412 > def selector(): ...
431
432     squek()
433     selector()
```

## Penjelasan

Fungsi ini digunakan sebagai *parent* dari fungsi fungsi lainnya.

## 2. Fungsi Database | def squek()

### Codes

```
15     def squek():
16 >         def createTable(): ...
34
35 >         def dataCount(): ...
42
43 >         def exampleData(): ...
48         createTable()
```

### Penjelasan

Fungsi ini digunakan sebagai *parent* dari fungsi yang berfokus ke pengaturan *database*, seperti membuat *table*, memasukkan data, dan menghitung data yang ada didalam *table* yg telah dibuat.

## 3. Fungsi Membuat Table | def createTable()

### Codes

```
16     def createTable():
17         cur.execute("""CREATE TABLE IF NOT EXISTS "category" (
18             "ID"    INTEGER,
19             "name"   TEXT NOT NULL,
20             PRIMARY KEY("ID" AUTOINCREMENT)
21         );""")
22
23         cur.execute("""CREATE TABLE IF NOT EXISTS "item" (
24             "ID"    INTEGER,
25             "name"   TEXT NOT NULL,
26             "catID"  INTEGER NOT NULL,
27             "stock"  INTEGER NOT NULL,
28             "price"  INTEGER NOT NULL,
29             "inputdate" TEXT NOT NULL,
30             PRIMARY KEY("ID" AUTOINCREMENT),
31             FOREIGN KEY("catID") REFERENCES [category] (ID)
32         );""")
33         exampleData()
```

### Penjelasan

Fungsi ini digunakan membuat tabel "*category*" dan tabel "*item*" (barang) yang berelasi lalu dia akan memanggil fungsi `exampleData()`.

#### 4. Fungsi menghitung data di table | def datacount()

##### Codes

```
35     def exampleData():
36         cur.execute("select count(*) as count from category")
37         for r2 in cur.fetchall():
38             count2 = r2[0]
39             if(count2 == 0):
40                 cur.execute("""insert into category (name)
41                             values('Beverage'), ('Food'),
42                             ('Cleaner'), ('Personal Care'), ('Other')
43                             """)
44                 conn.commit()
45     createTable()
```

##### Penjelasan

Fungsi ini digunakan untuk mengisi data ke dalam tabel "*category*" sebagai contoh data, pertama dia akan menghitung apakah tabel "*category*" sudah ada isinya atau tidak, jika setelah dilakukan perhitungan ternyata tidak ada data di dalam tabel "*category*" maka tabel tersebut akan diisi dengan data yang ada pada kode tersebut, sedangkan jika sudah ada maka akan melanjutkan fungsi selanjutnya.

#### 5. Fungsi menampilkan category | def showCat()

##### Codes

```
47     def showCat():
48         colCat = ['Id', 'Nama Kategori']
49         print("\n[i] Nama Kategori\n")
50         cur.execute("""select ID as 'Id',
51                     name as 'Nama Kategori' from category
52                     """)
53         newPT = pt()
54         newPT.field_names = colCat
55         for r in cur.fetchall():
56             newPT.add_row(r)
57         print(newPT)
```

##### Penjelasan

Fungsi ini digunakan untuk menampilkan tabel "*category*" dengan menggunakan module *PrettyTable*.

## 6. Fungsi Pengelola Item | `def item()`

### Codes

```
59  ✓   def item():
60      item.catid = 0
61      item.ct = 0
62  >   item_menu = ["[1] LIHAT DATA ITEM", ...
67
68  >   def showMenu(): ...
71
72  >   def checkTable(): ...
76
77  >   def showItem(): ...
90
91  >   def getCat(): ...
114
115  >   def createItem(): ...
135
136  >   def updateItem(): ...
198
199  >   def deleteItem(): ...
234
235  >   def inputan(): ...
268
269     inputan()
```

### Penjelasan

Fungsi ini digunakan sebagai parent dari fungsi yang berfokus ke pengelolaan barang atau item dengan basis *CRUD*, seperti menampilkan table "*item*", membuat data baru ke table "*item*", merubah data yang ada di dalam table "*item*", dan menghapus data dari table "*item*".

## 7. Fungsi Menampilkan Menu | `def showMenu()`

### Codes

```
68     def showMenu():
69         for i in item_menu:
70             print(i)
71
```

### Penjelasan

Fungsi ini digunakan untuk menampilkan list `item_menu`.

## 8. Fungsi Mengecek Isi Table Item | def checkTable()

### Codes

```
72  def checkTable():
73      cur.execute("select count(*) as count from item")
74  for r in cur.fetchall():
75      item.ct = r[0]
```

### Penjelasan

Fungsi ini digunakan untuk pengecekan apakah table "*item*" kosong atau tidak, dan fungsi ini akan digunakan pada fungsi inputan() nanti, lebih tepatnya ketika user memilih update ataupun delete sedangkan table "*item*" masih kosong sehingga akan terjadi error, dan dengan menggunakan fungsi ini akan mencegah user untuk membuka fungsi updateItem() dan fungsi deleteItem().

## 9. Fungsi Menampilkan Table Item | def showItem()

### Codes

```
77  def showItem():
78      colItems = ['Id', 'Nama Barang', 'Kategori', 'Stok', 'Harga', 'Tanggal Input']
79      print("\n[i] Daftar Barang\n")
80      cur.execute("""select
81                  item.ID as 'Id', item.name,
82                  category.name, item.stock,
83                  item.price,item.inputdate
84                  from item left join category
85                  on item.catID = category.ID
86                  """)
87
88      newPT = pt()
89      newPT.field_names = colItems
90      for r in cur.fetchall():
91          newPT.add_row(r)
92      print(newPT)
```

### Penjelasan

Fungsi ini digunakan menampilkan table "*item*" dengan menggunakan module *PrettyTable*, dalam query tersebut saya menggunakan *left join* karena saya memerlukan nama dari table "*category*" dengan merelasikan *item.catID* dengan *category.ID*.

## 10. Fungsi Mengambil Data dari Table Category | def getCat()

### Codes

```
91 def getCat():
92     print("\n[i] Untuk memasukkan kategori anda bisa mengisinya dengan Id maupun Nama dari kategori tersebut.")
93     inpCat = input("\n[=] Masukkan Id>Nama Kategori : ")
94     if(inpCat == ""):
95         return getCat()
96     elif(inpCat.isdigit()):
97         cur.execute("select * from category where ID=?", (inpCat,))
98         row = cur.fetchone()
99         if row == None:
100             print("\n[i] Tidak ada kategori yang ber-ID " + inpCat)
101             return getCat()
102         else:
103             item.catid = row[0]
104             print("\n[i] Kategori yang anda pilih adalah " + row[1])
105     else:
106         cur.execute("select * from category where name like ?", (inpCat,))
107         row = cur.fetchone()
108         if row == None:
109             print("\n[i] Tidak ada kategori yang bernama " + inpCat)
110             return getCat()
111         else:
112             item.catid = row[0]
113             print("\n[i] ID Kategori yang anda pilih adalah " + str(row[0]))
```

### Penjelasan

Fungsi ini digunakan untuk mengambil ID atau Nama dari table *"category"*, ketika user akan memasukkan barang ke table *"item"* dia bisa memasukkan ID atau Nama untuk mencari data yang tepat, dan supaya user dapat melakukan hal tersebut maka saya menambahkan 2 *if statements*, yaitu apakah user menggunakan ID, atau user menggunakan Nama untuk mencari data category yang tepat.

## 11. Fungsi Menambah Data ke Table Item | def createItem()

### Codes

```
115 def createItem():
116     print("\n[i] Anda akan memasukkan barang baru ke server.")
117     print("[i] Masukkan 0 atau Exit untuk kembali ke Menu Pilihan.")
118     inpName = input("[=] Masukkan Nama Barang : ")
119     showCat()
120     getCat()
121     inpStock = input("\n[=] Masukkan Stock Barang : ")
122     inpPrice = input("[=] Masukkan Harga Barang : ")
123     if(inpName == "" or inpStock == "" or inpPrice == ""):
124         return createItem()
125     elif(inpName == "0" or inpName == "exit" or inpName == "Exit" or inpName == "EXIT"):
126         return
127     elif(inpStock.isdigit() and inpPrice.isdigit()):
128         cur.execute("""insert into item (name, catID, stock, price, inputdate)
129             values(?, ?, ?, ?, ?)""",
130             [inpName, item.catid, inpStock, inpPrice, dt.date.today()],)
131         conn.commit()
132         print("\n[i] Data berhasil ditambahkan.")
133     else:
134         print("\n[!] Ada kesalahan ketika menginput data!")
135         return createItem()
```

### Penjelasan

Fungsi ini digunakan untuk menambahkan data ke dalam table "*item*", di dalam kode ini diharuskan untuk mengisi input harga dan stok dengan nomor, dan jika inputan tersebut diisi dengan karakter lain seperti huruf dan symbol maka user perlu mengisi inputan dari awal yaitu mengisi nama barangnya.

## 12. Fungsi Merubah Item Part 1 | def updateItem()

### Codes

```
133 def updateItem():
134     showItem()
135     print("\n[i] Anda akan mengubah beberapa atribut barang.")
136     print("[i] Jika anda tidak ingin memperbarui beberapa atribut, silahkan kosongi inputan tersebut.")
137     print("\n[i] Masukkan ID atau Nama barang untuk mengganti atribut barang.")
138     print("[i] Masukkan 0 atau Exit untuk kembali ke Menu Pilihan.")
139     inpID = input("[=] Masukkan ID atau Nama barang : ")
140
141     if(inpID == ""):
142         return updateItem()
143     elif(inpID == "0" or inpID == "exit" or inpID == "Exit" or inpID == "EXIT"):
144         return
145     elif(inpID.isdigit()):
146         cur.execute("""select item.ID, item.name, category.name , item.stock, item.price, category.id
147                        from item left join category on item.catID = category.ID where item.ID=?
148                        """, (inpID,))
149         getData = cur.fetchone()
150         if getData == None:
151             print("\n[i] Tidak ada barang yang ber-ID " + inpID)
152             return updateItem()
153         else:
154             inpID = getData[0]
155             print("\n[i] Barang yang anda pilih adalah " + getData[1])
156     else:
157         cur.execute("""select item.ID, item.name, category.name , item.stock, item.price, category.id
158                        from item left join category on item.catID = category.ID where item.name like ?
159                        """, (inpID,))
160         getData = cur.fetchone()
161         if getData == None:
162             print("\n[i] Tidak ada barang yang bernama " + inpID)
163             return updateItem()
164         else:
165             inpID = getData[0]
166             print("\n[i] ID Barang yang anda pilih adalah " + str(getData[0]))
```

### Penjelasan

Part 1 dari Fungsi ini digunakan untuk mengecek apakah inputan yang digunakan untuk mencari data merupakan sebuah ID atau Nama dari barang tersebut, lalu untuk setiap kemungkinan(ID atau Nama) tersebut akan menjalankan query yang disesuaikan pada *where clause*-nya.



### 13. Fungsi Merubah Item Part 2 | def updateItem()

#### Codes

```
168 | inpName = input("[=] Barang yang Bernama '" + getData[1] + "' Akan diganti Ke : ")
169 | print("\n[i] Barang ini Berkategori : " + getData[2])
170 | print("\n[=] Apakah anda ingin mengubah kategori barang ini?")
171 | yesnt = input("[=] Y/N : ")
172 | if(yesnt == "y" or yesnt == "Y"):
173 |     showCat()
174 |     getCat()
175 | elif(yesnt == "n" or yesnt == "N"):
176 |     item.catid = getData[5]
177 | inpStock = input("[=] Stok dari Barang ini adalah '" + str(getData[3]) + "' Akan diganti Ke : ")
178 | inpPrice = input("[=] Harga dari Barang ini adalah '" + str(getData[4]) + "' Akan diganti Ke : ")
179 | if(inpName == ""):
180 |     inpName = getData[1]
181 | if(inpStock == ""):
182 |     inpStock = getData[3]
183 | elif(not inpStock.isdigit()):
184 |     print("\n[!] Ada kesalahan ketika menginput data!")
185 |     return createItem()
186 | if(inpPrice == ""):
187 |     inpPrice = getData[4]
188 | elif(not inpPrice.isdigit()):
189 |     print("\n[!] Ada kesalahan ketika menginput data!")
190 |     return createItem()
191 | cur.execute("update item set name=?, catID=?, stock=?, price=? where id=?",
192 |             (inpName, item.catid, inpStock, inpPrice, inpID),)
193 | conn.commit()
194 | print("\n[i] Data berhasil dirubah.")
```

#### Penjelasan

Part 2 dari Fungsi ini digunakan untuk merubah data yang ada di table "*item*", user diperlukan untuk mengisi inputan berupa nama, kategori, stock, dan harga.

user bisa mengosongkan inputan untuk melewati apa yang tidak ingin diupdate, sebagai contoh user tidak ingin mengupdate harga, maka saat proses input stock user hanya perlu menekan enter untuk melewati proses inputan tersebut.

Khusus untuk inputan kategori, user perlu memilih untuk merubah kategori pada item itu dengan memilih **Y** atau **N**.

Ketika user menginputkan karakter yang tidak sesuai maka dia akan dikembalikan ke awal dari fungsi ini.

#### 14. Fungsi Menghapus Data dari Item | def deleteItem()

##### Codes

```
196 def deleteItem():
197     showItem()
198     print("[i] Masukkan ID atau Nama barang untuk menghapus.")
199     print("[i] Masukkan 0 atau Exit untuk kembali ke Menu Pilihan.")
200     inpID = input("[=] Masukkan ID atau Nama barang : ")
201     if(inpID == ""):
202         return deleteItem()
203     elif(inpID == "0" or inpID == "exit" or inpID == "Exit" or inpID == "EXIT"):
204         return
205     elif(inpID.isdigit()):
206         cur.execute("select * from item where ID=?", (inpID,))
207         getData = cur.fetchone()
208         if getData == None:
209             print("\n[i] Tidak ada barang yang ber-ID " + inpID)
210             return deleteItem()
211         else:
212             print("\n[i] Barang yang anda pilih adalah " + getData[1])
213     else:
214         cur.execute("select * from item where name like ?", (inpID,))
215         getData = cur.fetchone()
216         if getData == None:
217             print("\n[i] Tidak ada barang yang bernama " + inpID)
218             return deleteItem()
219         else:
220             print("\n[i] ID Barang yang anda pilih adalah " + str(getData[0]))
221
222     print("[i] Apakah anda yakin ingin menghapus data ini?")
223     inpUsure = input("[=] Y/N : ")
224     if(inpUsure == "y" or inpUsure == "Y"):
225         cur.execute("delete from item where id=?", (getData[0], ))
226         conn.commit()
227     elif(inpUsure == "n" or inpUsure == "N"):
228         return deleteItem()
229     else:
230         return deleteItem()
```

##### Penjelasan

Struktur Fungsi ini kurang lebihnya hampir sama dengan fungsi update, pertama user diperlukan untuk mengisi ID atau Nama dari barang, jika ID atau Nama barang sesuai dengan database lalu user akan ditanyai apakah user yakin untuk menghapus data ini, user bisa menjawab dengan **Y** atau **N**, lalu data akan dihapus dari *database*.

## 15. Fungsi memilih Menu Item | def inputan()

### Codes

```
231 |
232 | def inputan():
233 |     print('\n=====')
234 |     print('[i] Select Command')
235 |     print('=====')
236 |     showMenu()
237 |
238 |     inpSelect = input("\n[=] Pilih Menu : ")
239 |     checkTable()
240 |
241 |     if(inpSelect == '1'):
242 |         showItem()
243 |         return inputan()
244 |     elif(inpSelect == '2'):
245 |         createItem()
246 |         return inputan()
247 |     elif(inpSelect == '3'):
248 |         if(item.ct == 0):
249 |             print("\n[i] Tabel Data masih kosong!")
250 |         else:
251 |             updateItem()
252 |             return inputan()
253 |     elif(inpSelect == '4'):
254 |         if(item.ct == 0):
255 |             print("\n[i] Tabel Data masih kosong!")
256 |         else:
257 |             print("\n[i] Anda akan mengubah beberapa atribut barang.")
258 |             deleteItem()
259 |             return inputan()
260 |     elif(inpSelect == '5'):
261 |         return
262 |     else:
263 |         print("\n[!] Input Salah")
264 |         return inputan()
```

### Penjelasan

Fungsi ini digunakan untuk memilih menu, dari menu 1 sampai menu 5.

- menu 1 digunakan untuk menampilkan table item atau memanggil fungsi `showItem()`.
- menu 2 digunakan untuk mengisi table item atau memanggil fungsi `createItem()`.
- menu 3 digunakan untuk merubah data dari table item atau memanggil fungsi `updateItem()` jika table Item sudah ada isinya.
- menu 4 digunakan untuk menghapus data dari table item atau memanggil fungsi `deleteItem()` jika table Item sudah ada isinya.
- menu 5 untuk kembali ke menu utama yaitu fungsi `selector()`.

## 16. Fungsi Pengelola Category | def cat()

### Codes

```
268     def cat():
269         cat.ct = 0
270         cat_menu = ["[1] LIHAT DATA KATEGORI",
271                     "[2] TAMBAH DATA KATEGORI",
272                     "[3] EDIT DATA KATEGORI",
273                     "[4] HAPUS DATA KATEGORI",
274                     "[5] KELUAR DARI MENU KATEGORI"]
275
276 >     def showMenu(): ...
279
280 >     def checkTable(): ...
284
285 >     def createCat(): ...
297
298 >     def updateCat(): ...
331
332 >     def deleteCat(): ...
367
368 >     def inputan(): ...
401
402         inputan()
403
404 >     def showMenu(): ...
407
408 >     def selector(): ...
427
```

### Penjelasan

Fungsi ini digunakan sebagai *parent* dari fungsi-fungsi lainnya yang berfokus pada pengelolaan kategori.

#### 17. Fungsi menampilkan Menu Category | def showMenu()

##### Codes

```
276 def showMenu():
277     for i in cat_menu:
278         print(i)
279
```

##### Penjelasan

Fungsi ini digunakan untuk menampilkan list `cat_menu`.

#### 18. Fungsi Mengecek Isi Table Item | def checkTable()

##### Codes

```
280 def checkTable():
281     cur.execute("select count(*) as count from category")
282     for r in cur.fetchall():
283         cat.ct = r[0]
284
```

##### Penjelasan

Fungsi ini digunakan untuk pengecekan apakah table "*category*" kosong atau tidak, dan fungsi ini akan digunakan pada fungsi `inputan()` nanti, lebih tepatnya ketika user memilih update ataupun delete sedangkan table "*category*" masih kosong sehingga akan terjadi *error*, dan dengan menggunakan fungsi ini akan mencegah user untuk membuka fungsi `updateCat()` dan fungsi `deleteCat()`.

#### 19. Fungsi Menambah Data ke Table Category | def createCat()

##### Codes

```
285 def createCat():
286     print("\n[i] Anda akan mendaftarkan kategori baru ke server.")
287     print("[i] Masukkan 0 atau Exit untuk kembali ke Menu Pilihan.")
288     inpName = input("[=] Masukkan Nama Kategori : ")
289     if(inpName == ""):
290         return createCat()
291     elif(inpName == "0" or inpName == "exit" or inpName == "Exit" or inpName == "EXIT"):
292         return
293     else:
294         cur.execute("insert into category (name) values(?)", (inpName,))
295         conn.commit()
296         print("\n[i] Data berhasil ditambahkan.")
297
```

##### Penjelasan

Seperti Fungsi `createItem()` fungsi ini digunakan untuk menambah data ke dalam table "*category*".

## 20. Fungsi Merubah Category | def updateCat()

### Codes

```
298 def updateCat():
299     showCat()
300     print("\n[i] Masukkan ID atau Nama kategori untuk mengganti nama barang.")
301     print("[i] Masukkan 0 atau Exit untuk kembali ke Menu Pilihan.")
302     inpID = input("[=] Masukkan ID atau Nama kategori : ")
303
304     if(inpID == ""):
305         return updateCat()
306     elif(inpID == "0" or inpID == "exit" or inpID == "Exit" or inpID == "EXIT"):
307         return
308     elif(inpID.isdigit()):
309         cur.execute("select * from category where id=?", (inpID,))
310         getData = cur.fetchone()
311         if getData == None:
312             print("\n[i] Tidak ada kategori yang ber-ID " + inpID)
313             return updateCat()
314         else:
315             inpID = getData[0]
316             print("\n[i] Kategori yang anda pilih adalah " + getData[1])
317     else:
318         cur.execute("select * from category where name like ?", (inpID,))
319         getData = cur.fetchone()
320         if getData == None:
321             print("\n[i] Tidak ada kategori yang bernama " + inpID)
322             return updateCat()
323         else:
324             inpID = getData[0]
325             print("\n[i] ID Kategori yang anda pilih adalah " + str(getData[0]))
326
327     inpName = input("[=] Kategori yang Bernama '" + getData[1] + "' Akan diganti Ke : ")
328     cur.execute("update category set name=? where id=?", (inpName, inpID,))
329     conn.commit()
330     print("\n[i] Data berhasil dirubah.")
331
```

### Penjelasan

Fungsi ini digunakan untuk mengecek apakah inputan yang digunakan untuk mencari data merupakan sebuah ID atau Nama dari kategori tersebut, lalu untuk setiap kemungkinan(ID atau Nama) tersebut akan menjalankan query yang disesuaikan pada *where clause*-nya, untuk merubah data yang ada di table "*category*", user diperlukan untuk mengisi inputan nama.

## 21. Fungsi Menghapus Category | def deleteCat()

### Codes

```
332 def deleteCat():
333     showCat()
334     print("[i] Masukkan ID atau Nama kategori untuk menghapus.")
335     print("[i] Masukkan 0 atau Exit untuk kembali ke Menu Pilihan.")
336     inpID = input("[=] Masukkan ID atau Nama kategori : ")
337     if(inpID == ""):
338         return deleteCat()
339     elif(inpID == "0" or inpID == "exit" or inpID == "Exit" or inpID == "EXIT"):
340         return
341     elif(inpID.isdigit()):
342         cur.execute("select * from category where ID=?", (inpID,))
343         row = cur.fetchone()
344         if row == None:
345             print("\n[i] Tidak ada kategori yang ber-ID " + inpID)
346             return deleteCat()
347         else:
348             print("\n[i] Barang yang anda pilih adalah " + row[1])
349     else:
350         cur.execute("select * from category where name like ?", (inpID,))
351         row = cur.fetchone()
352         if row == None:
353             print("\n[i] Tidak ada kategori yang bernama " + inpID)
354             return deleteCat()
355         else:
356             print("\n[i] ID Kategori yang anda pilih adalah " + str(row[0]))
357
358     print("[i] Apakah anda yakin ingin menghapus data ini?")
359     inpUsure = input("[=] Y/N : ")
360     if(inpUsure == "y" or inpUsure == "Y"):
361         cur.execute("delete from category where id=?", (row[0],))
362         conn.commit()
363     elif(inpUsure == "n" or inpUsure == "N"):
364         return deleteCat()
365     else:
366         return deleteCat()
```

### Penjelasan

Struktur Fungsi ini kurang lebihnya sama dengan fungsi delete pada item, pertama user diperlukan untuk mengisi ID atau Nama dari barang, jika ID atau Nama barang sesuai dengan database lalu user akan ditanyai apakah user yakin untuk menghapus data ini, user bisa menjawab dengan **Y** atau **N**, lalu data akan dihapus dari *database*.

## 22. Fungsi memilih Menu Category | def inputan()

### Codes

```
368 def inputan():
369     print('\n=====')
370     print('[i] Select Command')
371     print('=====')
372     showMenu()
373
374     inpSelect = input("\n[=] Pilih Menu : ")
375     checkTable()
376
377     if(inpSelect == '1'):
378         showCat()
379         return inputan()
380     elif(inpSelect == '2'):
381         createCat()
382         return inputan()
383     elif(inpSelect == '3'):
384         if(cat.ct == 0):
385             print("\n[i] Tabel Data masih kosong!")
386         else:
387             updateCat()
388             return inputan()
389     elif(inpSelect == '4'):
390         if(cat.ct == 0):
391             print("\n[i] Tabel Data masih kosong!")
392         else:
393             print("\n[i] Anda akan mengubah beberapa atribut barang.")
394             deleteCat()
395             return inputan()
396     elif(inpSelect == '5'):
397         return
398     else:
399         print("\n[!] Input Salah")
400     return inputan()
```

### Penjelasan

Fungsi ini digunakan untuk memilih menu, dari menu 1 sampai menu 5.

- menu 1 digunakan untuk menampilkan table category atau memanggil fungsi `showCat()`.
- menu 2 digunakan untuk mengisi table category atau memanggil fungsi `createCat()`.
- menu 3 digunakan untuk merubah data dari table category atau memanggil fungsi `updateCat()` jika table category sudah ada isinya.
- menu 4 digunakan untuk menghapus data dari table category atau memanggil fungsi `deleteCat()` jika table category sudah ada isinya.
- menu 5 untuk kembali ke menu utama yaitu fungsi `selector()`.



### 23. Fungsi menampilkan Menu Utama | def showMenu()

#### Codes

```
404     def showMenu():
405         for i in main_menu:
406             print(i)
```

#### Penjelasan

Fungsi ini digunakan untuk menampilkan list `main_menu`.

### 24. Fungsi memilih Menu Utama | def selector()

#### Codes

```
408     def selector():
409         print('\n=====')
410         print('[i] Select Menu')
411         print('=====')
412         showMenu()
413
414         inpSelect = input("\n[=] Pilih Menu : ")
415
416         if(inpSelect == '1'):
417             item()
418             return selector()
419         elif(inpSelect == '2'):
420             cat()
421             return selector()
422         elif(inpSelect == '3'):
423             print("\n[i] Good Bye")
424         else:
425             print("\n[!] Input Salah")
426             return selector()
427
```

#### Penjelasan

Fungsi ini digunakan untuk memilih menu, dari menu 1 sampai menu 5.

- menu 1 digunakan untuk menampilkan Menu *Item* atau memanggil fungsi `item()`.
- menu 2 digunakan untuk menampilkan Menu *Category* atau memanggil fungsi `cat()`.
- menu 3 digunakan untuk keluar dari program.

## Screen-Shot Terminal

```
[ TOKO SUMBER JAYA ]
```

```
=====
[i] Select Menu
=====
```

```
[1] MASUK KE MENU ITEM
[2] MASUK KE MENU KATEGORI
[3] KELUAR DARI APP
```

```
[=] Pilih Menu : █
```

```
[=] Pilih Menu : 1
```

```
=====
[i] Select Command
=====
```

```
[1] LIHAT DATA ITEM
[2] TAMBAH DATA ITEM
[3] EDIT DATA ITEM
[4] HAPUS DATA ITEM
[5] KELUAR DARI MENU ITEM
```

```
[=] Pilih Menu : █
```

```
[=] Pilih Menu : 1
```

```
[i] Daftar Barang
```

Id	Nama Barang	Kategori	Stok	Harga	Tanggal Input
1	Aramasa	Other	50	10000	2022-01-09
2	Asakusa	Other	100	8000	2022-01-09
4	Awakasa	Other	50	5000	2022-01-09
5	Ayakasa	Other	50	8000	2022-01-09

```
=====
[i] Select Command
=====
```

```
[1] LIHAT DATA ITEM
[2] TAMBAH DATA ITEM
[3] EDIT DATA ITEM
[4] HAPUS DATA ITEM
[5] KELUAR DARI MENU ITEM
```

```
[=] Pilih Menu : █
```

[=] Pilih Menu : 2

[i] Anda akan memasukkan barang baru ke server.

[i] Masukkan 0 atau Exit untuk kembali ke Menu Pilihan.

[=] Masukkan Nama Barang :

[i] Anda akan memasukkan barang baru ke server.

[i] Masukkan 0 atau Exit untuk kembali ke Menu Pilihan.

[=] Masukkan Nama Barang : Arumisa

[i] Nama Kategori

```
+---+-----+
| Id | Nama Kategori |
+---+-----+
| 1  | Beverage      |
| 2  | Food           |
| 3  | Cleaner        |
| 4  | Personal Care  |
| 5  | Other          |
+---+-----+
```

[i] Untuk memasukkan kategori anda bisa mengisinya dengan Id maupun Nama dari kategori tersebut.

[=] Masukkan Id>Nama Kategori : 5

[i] Kategori yang anda pilih adalah Other

[=] Masukkan Stock Barang : 50

[=] Masukkan Harga Barang : 8500

[i] Data berhasil ditambahkan.

=====

[i] Select Command

=====

[1] LIHAT DATA ITEM

[2] TAMBAH DATA ITEM

[3] EDIT DATA ITEM

[4] HAPUS DATA ITEM

[5] KELUAR DARI MENU ITEM

[=] Pilih Menu :

[=] Pilih Menu : 3

[i] Daftar Barang

Id	Nama Barang	Kategori	Stok	Harga	Tanggal Input
1	Aramasa	Other	50	10000	2022-01-09
2	Asakusa	Other	100	8000	2022-01-09
4	Awakasa	Other	50	5000	2022-01-09
5	Ayakasa	Other	50	8000	2022-01-09
12	Arumisa	Other	50	8500	2022-01-09

[i] Anda akan mengubah beberapa atribut barang.

[i] Jika anda tidak ingin memperbarui beberapa atribut, silahkan kosongi inputan tersebut.

[i] Masukkan ID atau Nama barang untuk mengganti atribut barang.

[i] Masukkan 0 atau Exit untuk kembali ke Menu Pilihan.

[=] Masukkan ID atau Nama barang : █

[i] Masukkan ID atau Nama barang untuk mengganti atribut barang.

[i] Masukkan 0 atau Exit untuk kembali ke Menu Pilihan.

[=] Masukkan ID atau Nama barang : arumisa

[i] ID Barang yang anda pilih adalah 12

[=] Barang yang Bernama 'Arumisa' Akan diganti Ke :

[i] Barang ini Berkategori : Other

[=] Apakah anda ingin mengubah kategori barang ini?

[=] Y/N : n

[=] Stok dari Barang ini adalah '50' Akan diganti Ke :

[=] Harga dari Barang ini adalah '8500' Akan diganti Ke : 9500

[i] Data berhasil dirubah.

[=] Pilih Menu : 4

[i] Anda akan mengubah beberapa atribut barang.

[i] Daftar Barang

Id	Nama Barang	Kategori	Stok	Harga	Tanggal Input
1	Aramasa	Other	50	10000	2022-01-09
2	Asakusa	Other	100	8000	2022-01-09
4	Awakasa	Other	50	5000	2022-01-09
5	Ayakasa	Other	50	8000	2022-01-09
12	Arumisa	Other	50	9500	2022-01-09

[i] Masukkan ID atau Nama barang untuk menghapus.

[i] Masukkan 0 atau Exit untuk kembali ke Menu Pilihan.

[=] Masukkan ID atau Nama barang :

[i] Masukkan ID atau Nama barang untuk menghapus.

[i] Masukkan 0 atau Exit untuk kembali ke Menu Pilihan.

[=] Masukkan ID atau Nama barang : 12

[i] Barang yang anda pilih adalah Arumisa

[i] Apakah anda yakin ingin menghapus data ini?

[=] Y/N : y

[i] Data berhasil dihapus!

[=] Pilih Menu : 5

[i] Select Menu

[1] MASUK KE MENU ITEM

[2] MASUK KE MENU KATEGORI

[3] KELUAR DARI APP

[=] Pilih Menu :

[=] Pilih Menu : 2

[i] Select Command

[1] LIHAT DATA KATEGORI

[2] TAMBAH DATA KATEGORI

[3] EDIT DATA KATEGORI

[4] HAPUS DATA KATEGORI

[5] KELUAR DARI MENU KATEGORI

[=] Pilih Menu :

[=] Pilih Menu : 2

[i] Anda akan mendaftarkan kategori baru ke server.

[i] Masukkan 0 atau Exit untuk kembali ke Menu Pilihan.

[=] Masukkan Nama Kategori : Elektronik

[i] Data berhasil ditambahkan.

[=] Pilih Menu : 3

[i] Nama Kategori

+---+-----+	
Id	Nama Kategori
+---+-----+	
1	Beverage
2	Food
3	Cleaner
4	Personal Care
5	Other
9	Elektronik
+---+-----+	

[i] Masukkan ID atau Nama kategori untuk mengganti nama barang.

[i] Masukkan 0 atau Exit untuk kembali ke Menu Pilihan.

[=] Masukkan ID atau Nama kategori : elektron

[i] Tidak ada kategori yang bernama elektron

[i] Nama Kategori

+---+-----+	
Id	Nama Kategori
+---+-----+	
1	Beverage
2	Food
3	Cleaner
4	Personal Care
5	Other
9	Elektronik
+---+-----+	

[i] Masukkan ID atau Nama kategori untuk mengganti nama barang.

[i] Masukkan 0 atau Exit untuk kembali ke Menu Pilihan.

[=] Masukkan ID atau Nama kategori : elektronik

[i] ID Kategori yang anda pilih adalah 9

[=] Kategori yang Bernama 'Elektronik' Akan diganti Ke : Electronic

[i] Data berhasil dirubah.

```
=====
[i] Select Command
=====
[1] LIHAT DATA KATEGORI
[2] TAMBAH DATA KATEGORI
[3] EDIT DATA KATEGORI
[4] HAPUS DATA KATEGORI
[5] KELUAR DARI MENU KATEGORI

[=] Pilih Menu : 4

[i] Anda akan mengubah beberapa atribut barang.

[i] Nama Kategori

+---+-----+
| Id | Nama Kategori |
+---+-----+
| 1  | Beverage      |
| 2  | Food           |
| 3  | Cleaner        |
| 4  | Personal Care  |
| 5  | Other          |
| 9  | Electronic     |
+---+-----+

[i] Masukkan ID atau Nama kategori untuk menghapus.
[i] Masukkan 0 atau Exit untuk kembali ke Menu Pilihan.
[=] Masukkan ID atau Nama kategori : 9

[i] Barang yang anda pilih adalah Electronic
[i] Apakah anda yakin ingin menghapus data ini?
[=] Y/N : y
[i] Data berhasil dihapus!
```

```
=====
[i] Select Command
=====
[1] LIHAT DATA KATEGORI
[2] TAMBAH DATA KATEGORI
[3] EDIT DATA KATEGORI
[4] HAPUS DATA KATEGORI
[5] KELUAR DARI MENU KATEGORI

[=] Pilih Menu : 5

=====
[i] Select Menu
=====
[1] MASUK KE MENU ITEM
[2] MASUK KE MENU KATEGORI
[3] KELUAR DARI APP

[=] Pilih Menu : 3

[i] Good Bye
PS C:\Users\Allan\Desktop\Univ\PY and DB\responsi uas> |
```