

Tablolar

	calisan_id [PK] integer	calisan_isim character varying (255)	calisan_tel integer	calisan_adres character varying (255)	calisan_bolum_no integer
1	9711	Enes Köseoğlu	3435787	Bayrampaşa	3
2	2684	İsa Gül	2533300	Güngören	2
3	9829	Hakan Adıyaman	1852460	Bahçelievler	8
4	7799	Rahim Albayrak	6686642	Pendik	1
5	6059	Selin Yılmaz	9172843	Esenler	5
6	4803	Aleyna Türk	4450838	Kadıköy	6
7	3150	Muhammet Koç	4679283	Şişli	9
8	7937	Mert Deniz	5909140	Bakırköy	7
9	3175	Asya Yener	6062311	Mecidiyeköy	4
10	7151	Sena Ekiz	1369268	Ümraniye	10
11	1001	Abdullah Aslan	1329228	Gebze	3
12	1002	Merdi Korutürk	1341648	Darıca	3
13	1032	Ülfet Arsoy	1345678	Beşiktaş	3
14	1052	Özlem Dumanlı	1361138	İçmeler	4
15	1042	Bilgin Gülen	1361518	Esenler	4
16	1190	Nirgül Şama	1369878	Üsküdar	4
17	9111	Enes Köseoğlu	3435787	Bayrampaşa	3
18	9811	Enes Köseoğlu	3435787	Bayrampaşa	3

	ureticci_id [PK] integer	ureticci_isim character varying (255)	ureticci_tel integer	ureticci_adres character varying (255)	toplam_zeytin integer	toplam_zeytinyagi integer
1	9092	Mesut Yağcı	9225789	İzmir	250	0
2	8470	Evren Sezikli	1007500	Balıkesir	300	0
3	2196	Güral Dağışan	3953201	Çanakkale	200	0
4	5232	Damla Şenel	9445967	Bursa	450	0
5	4057	Kemal Tosun	1250270	Kütahya	225	0
6	6321	Celal Tan	9462585	Mersin	150	0
7	4444	Müge Kısa	3468598	Manisa	545	1215
8	2257	Aycan Karabay	8160180	Aydın	100	50
9	6229	Ayşe Kırbız	2154059	Uşak	425	150
10	4538	Sena Karatürk	8042053	Muğla	400	150

	musteri_id [PK] integer	musteri_isim character varying (255)	musteri_tel integer	musteri_adres character varying (255)	toplam_siparis integer
1	5710	Renan Salik	8244538	Burdur	20
2	9403	Hatice Karataş	9554501	Adıyaman	15
3	4588	Benan Üzen	8816331	Çorum	45
4	4657	Feza Karaer	7303221	Denizli	50
5	5264	Hayati Doğru	7580482	Gümüşhane	33
6	7516	Mustafa Ürel	3165596	Afyonkarahisar	27
7	4937	Hasibe Duysak	2206544	Konya	55
8	1573	Ali Bağcı	5300194	Malatya	60
9	9201	Veli Özdoğan	6736745	Yalova	32
10	4110	Didem Çipe	5410904	Mardin	117

	siparis_id [PK] integer	musteri_no integer	miktar integer	tarih timestamp without time zone
1	16	5710	20	2023-01-22 10:45:12
2	17	9403	15	2023-01-21 07:12:32
3	18	4588	45	2023-01-30 11:55:52
4	19	4657	50	2023-01-26 17:33:02
5	20	5264	33	2023-01-20 10:45:12
6	21	7516	27	2023-01-22 12:28:27
7	22	4937	55	2023-01-25 21:45:45
8	23	1573	60	2023-01-21 20:19:37
9	24	9201	32	2023-01-27 18:16:07
10	25	4110	44	2023-01-28 13:26:47

	kayit_id [PK] integer	uretilci_no integer	islem character varying (255)	miktar integer	tarih timestamp without time zone
1	22	2257	getirdi	100	2023-01-16 12:50:25
2	23	9092	getirdi	250	2023-01-12 13:22:35
3	24	8470	getirdi	300	2023-01-06 14:20:15
4	25	6229	getirdi	125	2023-01-01 17:32:12
5	26	4538	getirdi	400	2023-01-05 21:18:44
6	27	2196	getirdi	200	2023-01-14 22:37:43
7	28	5232	getirdi	450	2023-01-07 20:41:30
8	29	4057	getirdi	225	2023-01-09 19:07:55
9	30	6321	getirdi	150	2023-01-10 09:17:26
10	31	4444	getirdi	325	2023-01-02 10:04:22
11	32	5232	getirdi	150	2023-01-10 09:17:26
12	33	6229	getirdi	325	2023-01-02 10:04:22
13	34	4538	goturdu	240	2022-12-22 13:55:32
14	35	4538	goturdu	325	2022-03-02 15:12:21
15	36	4444	getirdi	180	2022-01-16 11:42:33
16	37	4444	goturdu	240	2022-12-22 13:55:32
17	38	4444	goturdu	325	2022-04-09 13:19:42
18	39	4444	goturdu	325	2022-02-24 22:21:52
19	40	4444	getirdi	40	2022-04-02 21:04:12

	bolum_id [PK] integer	bolum_adi character varying (255)	calisan_sayisi integer
1	2	Depolama	1
2	8	Sınıflandırma	1
3	1	Sıkma	1
4	5	Satış	1
5	6	Nakliyat	1
6	9	Yıkama	1
7	7	Tedarik	1
8	10	Fermantasyon	1
9	4	Salamura	4
10	3	Paketleme	6

Arayüzden Insert, Delete, Update İşlemleri

Aşağıdaki Python kodu ile arayüz üzerinden alınan bilgiler ile tablolar üzerinde işlem yapılmaktadır. "tablo" değişkeni tablo ismini, "extra" kısmı ise gerekli olan girdilerin kullanıcıdan alındığı kısımdır.

```
if islem == "Insert":  
    query = f"INSERT INTO {tablo} {extra}"  
elif islem == "Update":  
    query = f"UPDATE {tablo} {extra}"  
else:  
    query = f"DELETE FROM {tablo} {extra}"
```

İşlem	Tablo	Ek	
Insert	calisanlar	values(9999,'Ahmet Koç', 555555, 'Kars', 3)	Çalıştır
Durum: Başarılı			

Sipariş Ekle		
Müşteri ID:	4110	Çalıştır
Miktar:	20	Durum: Başarılı
Tarih ve Saat:	2023-01-18 10:10	

Kayıt Ekle		
Üretici ID:	9092	Çalıştır
İşlem:	Zeytin getirdi	Durum: Başarılı
Miktar:	100	
Tarih ve Saat:	2023-01-18 10:10	

Arayüz Üzerinden Select İşlemi

Aşağıdaki kod parçası ile arayüz üzerinden seçilen tablonun içeriği ekrana yazdırılmaktadır.

```
if len(extra)==1:
    cur.execute(f"SELECT {cols} FROM {tablo}")
else:
    cur.execute(f"SELECT {cols} FROM {tablo} WHERE {extra}")
```

View ile Sorgu

Arayüz üzerinden alınan değerden fazla çalışanı olan bölümler ekrana yazdırılırken "view" kullanılmaktadır.

```
param = textbox_sorgu_param.get("1.0", END)
create_view = f"""
    CREATE OR REPLACE VIEW view1 AS
    SELECT * FROM bolumler
    WHERE calisan_sayisi > {int(param)}
    ""
cur.execute(create_view)
```

Sequence

```
-- Kayitlarin tutuldugu tabloda kayıt numarasini otomatik olarak 1 arttıran
sequence
CREATE SEQUENCE kayıt_id_seq START 1 INCREMENT 1;

-- Siparislerin tutuldugu tabloda siparis numarasini otomatik olarak 1
arttıran sequence
CREATE SEQUENCE siparis_id_seq START 1 INCREMENT 1;
```

Union Kullanımı

Arayüzden alınan değerden fazla zeytin getiren ve zeytinyağı götüren üreticiler UNION kullanılarak listelenir.

```
param = textbox_sorgu_param.get("1.0", END)
query = f"""
    SELECT uretici_isim, toplam_zeytin, toplam_zeytinyagi FROM ureticiler
    WHERE toplam_zeytin > {int(param)}
    UNION
    SELECT uretici_isim, toplam_zeytin, toplam_zeytinyagi FROM ureticiler
    WHERE toplam_zeytinyagi > {int(param)}
    ""
```

```
cur.execute(query)
```

Aggregate Kullanımı

Girilen değerden fazla sipariş veren müşteriler siparişler tablosunda “group” by ve “having” yapılarak bulunur.

```
param = textbox_sorgu_param.get("1.0", END)
query = f"""
    SELECT muster_i_no, COUNT(siparis_id)
    FROM siparisler
    GROUP BY muster_i_no
    HAVING COUNT(siparis_id) > {int(param)}
    """
cur.execute(query)
```

Fonksiyon Kullanımı

Aşağıdaki kodlar ile veritabanında bulunan fonksiyonlar arayüzden alınan değer ile çalıştırılmaktadır.

```
def execute_fonk():
    fonk = clicked_fonk.get()
    param = int(textbox_param.get("1.0", END))
    if "Üreticinin getirdiği" in fonk:
        cur.execute(f"SELECT * FROM uretici_toplam({param})")
        res = cur.fetchone()
    elif "Bir bölümdeki" in fonk:
        cur.execute(f"SELECT * FROM bolum_calisan_sayisi({param})")
        res = cur.fetchone()
    else:
        cur.execute(f"SELECT * FROM fazla_olan({param})")
        res = cur.fetchone()
    textbox_fonk_sonuc.delete("1.0", END)
    textbox_fonk_sonuc.insert(INSERT, res)
```

Fonksiyon

Üreticinin getirdiği toplam zeytin ağırlığı ve aldığı toplam zeytinyağı hacmi ▼

Parametre: 6229

Çalıştır

Sonuç: 425 150

Fonksiyon

Bir bölümdeki toplam çalışan sayısı ▾

Parametre: 3

Çalıştır

Sonuç: 6

Fonksiyon

Getirilen toplam zeytin miktarı belirli bir değerden fazla olan üretici sayısı ▾

Parametre: 200

Çalıştır

Sonuç: 7

Trigger Tetikleme

Aşağıdaki trigger ile yeni bir sipariş kaydı verildiğinde müşterinin toplam sipariş miktarı otomatik arttırılmaktadır. Aynı durum üreticilerin getirdiği zeytin ve götürdüğü zeytinyağı miktarı için de yapılmaktadır.

```
-- Yeni siparis eklendiginde musterinin toplam siparis miktarini guncelleyen
trigger
CREATE OR REPLACE FUNCTION yeni_siparis_fonk()
RETURNS TRIGGER AS $$
BEGIN
    UPDATE musteriler SET toplam_siparis=toplam_siparis+new.miktar
    WHEREmusteri_id=new.musteri_no ;
RETURN NEW;
END;
$$ LANGUAGE 'plpgsql';

-- Yeni kayıt eklendiginde kayıt turune gore ureticinin toplam degerleri
guncelleyen trigger
CREATE OR REPLACE FUNCTION yeni_kayit_fonk()
RETURNS TRIGGER AS $$
BEGIN
    IF new.islem LIKE '%getirdi%' THEN
        UPDATE ureticiler SET toplam_zeytin=toplam_zeytin+new.miktar
        WHEREuretici_id=new.uretici_no ;
    ELSE
        UPDATE ureticiler SET toplam_zeytinyagi=toplam_zeytinyagi+new.miktar
        WHEREuretici_id=new.uretici_no ;
    END IF;
RETURN NEW;
END;
$$ LANGUAGE 'plpgsql';
```

```
    END IF;  
RETURN NEW;  
END;  
$$ LANGUAGE 'plpgsql';
```