SELECT InvoiceId, BillingCountry, total FROM invoices WHERE total >(SELECT round(avg(total)) FROM invoices);

SELECT InvoiceId, BillingCountry, total FROM invoices WHERE BillingCountry='Germany' and total >(SELECT round(avg(total)) FROM invoices );

--

SELECT \* FROM tracks WHERE AlbumId=(SELECT AlbumId FROM albums WHERE Title='Killers');

--

SELECT tracks.TrackId, tracks.Name, albums.AlbumId, artists.Name

FROM albums

INNER JOIN tracks on tracks.AlbumId=albums.AlbumId

INNER JOIN artists on albums.ArtistId=artists.ArtistId

WHERE albums.Title='Killers';

--

SELECT \* FROM albums WHERE AlbumId IN (SELECT AlbumId FROM tracks WHERE Composer='AC/DC');

--coklu sutun dondurmek icin

SELECT \* FROM albums WHERE AlbumId IN(SELECT AlbumId FROM tracks WHERE Composer IN('AC/DC','Jerry Cantrell','Audioslave/Chris Cornell'));

--join ile

SELECT albums.AlbumId, albums.Title, albums.ArtistId, tracks.Composer FROM albums

INNER JOIN tracks on tracks.AlbumId=albums.AlbumId

WHERE tracks.Composer IN('AC/DC','Jerry Cantrell','Audioslave/Chris Cornell');

--Tüm artistler albüm çıkartmış mı bakalım

SELECT artists.Name as artis\_ismi, a.Title as album\_ismi FROM artists

LEFT JOIN albums as a on artists.ArtistId=a.ArtistId WHERE album\_ismi IS NULL;

--null olanları saydıralım

SELECT count(\*) null\_olan FROM artists

LEFT JOIN albums as a on artists.ArtistId=a.ArtistId WHERE a.Title IS NULL;

SELECT count(artists.Name)-count(albums.Title)FROM artists LEFT JOIN albums on artists.ArtistId=albums.ArtistId;

--

SELECT artists.Name as artis\_ismi, a.Title as album\_ismi FROM artists

INNER JOIN albums as a on artists.ArtistId=a.ArtistId GROUP BY artists.Name;

--album ismi tek oldugu icin gruplama yapmadi

SELECT artists.Name as artis\_ismi, a.Title as album\_ismi FROM artists

INNER JOIN albums as a on artists.ArtistId=a.ArtistId GROUP BY a.Title;

--her sanatcının kac albumu var

SELECT artists.Name as artis\_ismi, count(albums.Title) FROM artists

INNER JOIN albums as albums on artists.ArtistId=albums.ArtistId GROUP BY artists.Name;

SELECT artists.Name as artis\_ismi, count(albums.Title) FROM artists

INNER JOIN albums as albums on artists.ArtistId=albums.ArtistId GROUP BY artists.Name HAVING count(albums.Title)>5;

--

SELECT Composer, count(Composer) FROM tracks where Composer like ('A%') GROUP BY Composer HAVING count(Composer)>5;

SELECT InvoiceId, InvoiceDate, BillingCountry FROM invoices WHERE InvoiceDate>'2010-12-25 00:00:00' GROUP BY BillingCountry HAVING InvoiceId>186;

--2011 den sonra fatura toplamı ulkelere gore 50 den büyük olanların tum bilgisini getir

SELECT BillingCountry, sum(total) as toplam, \* FROM invoices WHERE InvoiceDate>'2011-01-01 00:00:00' GROUP BY BillingCountry HAVING toplam>50;

CREATE TABLE "yeni"(id INTEGER, isim TEXT, soyisim TEXT);

CREATE TABLE if not EXISTS "yeni"(id INTEGER, isim TEXT, soyisim TEXT);

DROP TABLE yeni;

DROP TABLE yenitablo;

CREATE TABLE "izinler"(id INTEGER, employe\_id INTEGER, bas\_tarihi date, bitis\_tarihi date);

--

SELECT izinler.id, carsalesEmployees.id, carsalesEmployees.name FROM izinler inner JOIN carsalesEmployees on izinler.employe\_id=carsalesEmployees.id;

--tatil planı

CREATE TABLE "tatil\_plani"(tatil\_id INTEGER, calisan\_id INTEGER , sehir TEXT, otel\_ismi TEXT, tatil\_uzunlugu INTEGER, butce INTEGER);

--

INSERT into tatil\_plani (tatil\_id, calisan\_id, sehir, otel\_ismi, tatil\_uzunlugu, butce) VALUES (1,3, "Antalya", "IBIS", 7, 10000);

--

INSERT INTO izinler(id, emp\_id, bas\_tarihi, bitis\_tarihi) VALUES(1, 7, '2011-01-01', '2011-02-02');