NAMA:NIRMALA

NIM:200250502049

KELAS:TEKNIK INFORMATIKA B

**A. SQL tutorial**

1. SQL Home

SQL adalah bahasa standar untuk menyimpan, memanipulasi dan mengambil data dalam database.

1. SQL Intro

SQL adalah singkatan dari Structured Query Language SQL memungkinkan Anda mengakses dan memanipulasi database SQL menjadi standar American National Standards Institute (ANSI) pada 1986, dan International Organization for Standardization (ISO) pada 1987 .

**Contoh:**

SELECT \* FROM customers;

1. SQL Syntax

Database paling sering berisi satu atau lebih tabel. Setiap tabel diidentifikasi dengan nama (misalnya "Pelanggan" atau "Pesanan"). Tabel berisi catatan (baris) dengan data.

**Contoh:**

SELECT \* FROM Buyers;

1. SQL Select

SQL Select adalah query yang bisa digunakan untuk menampilkan maupun mengambil data pada sebuah tabel, data yang diambil tidak hanya terbatas pada satu jenis.

Syntas dasar:

SELECT column\_name(s)

FROM table\_name

Atau

SELECT\*FROM table\_name

Contoh :

SELECT nama, email

FROM tabeldata

1. SQL Select Distinct

SQL Select Distinict adalah sql yang digunakan untuk meilih data yang berbeda (menghilanhkan duplikasi)dari sebuah table database.

Syntax dasar:

SELECT DISTINCT column\_name(s)

FROM table\_name

Contoh :

SELECT DISTINCT kota

FROM tabeldata

1. SQL WhereSQL And, Or, Not

SQL WhereSQL And, Or, Not merupakan perintah dasar SQL yang biasanya di kombinasikan dengan perintah WHERE. Ketiganya digunakan untuk memfilter record berdasarkan suatu kondisi, operator AND akan menampilkan record apabila semua kondisi bernilai TRUE, operator OR akan menampilkan record apabila salah satu kondisi bernilai TRUE, sedangkan operator NOT akan menampilkan record apabila semua kondisi bernilai FALSE.

Syntax AND :

SELECT kolom1, kolom2,….

FROM nama\_tabel

WHERE kondisi1 AND kondisi2 AND kondisi3…;

Contoh AND :

SELECT nis, nama FROM siswa WHERE

alamat=’Jakarta’ AND

tahun\_lahir=’2000’

Syntax OR:

SELECT kolom1, kolom2,…

FROM nama\_table

WHERE kondisi1 OR kondisi2 OR kondisi3

Contoh OR:

SELECT nis, nama FROM siswa WHERE

Alamat=”jakarta’ OR alamat=’Bandung’

Syntax NOT:

SELECT kolom1, kolom2,…

FROM nama\_table WHERE NOT kondisi

Contoh NOT :

SELECT nis, nama

FROM siswa WHERE

NOT alamat= ‘Jakarta’

1. SQL Order By

SQL order by di gunakan untuk mengurutkan data berdasarkan kolom (filed) tertentu. Secara default, urutan tersusun secara ascending (urut kecil ke besar). Anda dapat mengubahnya menjadi discending (urut besar ke kecil) dengan menambah perintah DESC.

Syntax dasar:

SELECT column\_name(s)

FROM table\_name

ORDER BY column\_name(s) ASCIDESSC

Contoh :

SELECT\*FROM tabeldata ORDER BY nama

1. SQL Insert Int

Dalam SQL, perintah INSERT INTO merupakan perintah dasar SQL bagian dari perintah untuk DML.

Syntax dasar :

INSERT INTO nama\_tabel VALUES

(nilai1, nilai2, nilai3,…..)

Contoh :

INSERT INTO siswa VALUES (‘12345’, Abdul’);

1. SQL Null Values

SQL Null Values adalah bidang tanpa nilai. Jika bidang dalam tabel adalah opsional, dimungkinkan untuk menyisipkan catatan baru atau menperbaruhi catatan tanpa menambahkan nilai ke bidang ini. kemudian, bidang tersebut akan disimpan dengan nilai NULL.

Sintaks IS NULL

SELECT column\_names

FROM table\_name

WHERE column\_name IS NOT NULL

Operator IS NULL digunakan untuk menguji nilai kosong (nilai null). Sql berikut mencantumkan semua pelanggan dengan null di bidang “ALAMAT”:

Contoh :

SELECT customerName, contactName, Address

FROM customers

WHERE Address IS NUL;

1. SQL Update SQL Delete

SQL Update digunakan untuk mengubah/menperbaruhi datadi tabel database.

Syntax dasar:

UPDATE table\_name

SET column1=value,column2=value,…

WHERE some\_column=some\_value

Contoh :

UPDATE tabeldata SET

Email=’redaksi@kuliahkomputer.Com’,

Kota=’kediri’

WHERE id = 1

SQL Delete digunkan untuk menghapus data di tabel database. Tambahkan perintah Where untuk memfilter data-data tertentu yang akan di hapus. Jika tanpa perintah Where, maka seluruh data dalam tabel akan terhapus

Syntax dasar :

DELETE FROM table\_name

WHERE some\_column=some\_value

Contoh

DELETE FROM tabeldata WHERE id =1

1. SQL Select Top

SQL Select Top digunakan untuk menentukan jumlah record yang akan di tampilkan. Kalusa SELECT TOP berguna pad atabel dengan row data yang banyak dengan ribuan record data. Menampilkan sejumlah besar record data dapat berdampak pada kinerja database.

Syntax :

SELECT nama\_kolom(s)

FROM nama\_table

WHERE kondisi

LIMIT nomor;

Contoh :

SELECT \*FROM Customers

LIMIT 3;

1. SQL Min and Max

SQL Max digunakan untuk mendapatkan nilai terbesar dari data-data yang ada.

Syntax dasar :

SELECT MAX (column\_name) FROM table\_name

Contoh :

SELECT MAX(harga) AS Harga\_termahal FROM order

SQL Min digunakan untuk mendapatkan nilai terkecil dari data-data yang ada.

Syntax dasar :

SELECT MIN (column\_name) FROM table\_name

Contoh:

SELECT MIN(harga) AS Harga\_termurah FROM order

1. SQL Count, Avg, Sum

SQL Count Digunakan untuk menghitung jumlah (cacah) suatu data.

Syntax dasar :

SELECT COUNT (column\_name) FROM table\_name

Contoh :

SELECT COUNT(id) AS Jumlah\_tamu FROM tabeldata

SQL Avg digunakan untuk menghitung nilai-rata-rata dari suatu data.

Syntax dasar :

SELECT AVG (column\_name) FROM table\_name

Contoh :

SELECT AVG(harga) AS Harga\_rata2FROM order

SQL Sum digunakan untuk mendapatkan nilai total penjumlahan dari data-data yang ada.

Syntax dasar :

SELECT SUM (column\_name) FROM table\_name

Contoh :

SELECT SUM(harga) AS Harga\_total FROM order

1. SQL Like

SQL Like digunakan bersama dengan perintah Where, untuk proses pencarian data dengan spesifikasi tertentu.

Syntax dasar:

SELECT column\_name(s)

FROM table\_name

WHERE column\_name LIKE pattern

Contoh :

SELECT\*FROM tabeldata WHERE nama

LIKE’z%’

Ket : contoh di atas digunakan untuk pencarian berdasarkan kolom nama yang berhuruf depan”z”

SELECT\*FROM tabeldata WHERE nama

LIKE’z%’

Ket : contoh di atas digunakan untuk pencarian berdasarkan kolom nama yang berhuruf belakang”z”

1. SQL Wildcards

SQL Wildcards di gunakan untuk menggantikan satu atau lebih krakter dalam sebuah string. Wildcards di gunakan dengan operator LIKE . Operator LIKE digunakan dalam klausa WHERE untuk mencari pola tertentu dalam kolom.

Contoh :

SELECT \* FROM Customer

WHERE kota LIKE ‘ber%’:

1. SQL In

SQL In digunakan untuk pencarian data menggunakan lebih dari satu filter pada perintah Where

Syntas dasar:

SELECT column-name(s)

FROM table\_name

WHERE column\_name IN (value1, value2,..)

Contoh :

SELECT\*FROM tabeldata WHERE kota IN

(‘kediri’,malang’)

BE

1. SQL Between

SQL Between digunakan untuk menentukan jangkauan pencarian.

Syntax dasar:

SELECT column\_name(s)

FROM table-name

WHERE column-name

BETWEEN value1 AND value2

Contoh :

SELECT\*FROM\*tabeldata WHERE id

BETWEEN 15 and 45

Ket : contoh di atas digunakanuntuk mencari data yang memiliki nomor idantara 15 dan 45.

1. SQL Aliases

As memungkinkan anda untuk menberi nama ulang/nama alias/nama baru pada sebuah kolom atau tabel tanpa mengubah nama aslinya. Beberapa programmer sering menggunakan fitur AS ini untuk menbantu mereka dalam mengenali sebuah tabel atau kolom serta menpermudah dalam penulisan query secara ringkas.

Syntax dasar :

SELECT nama\_kolom as nama\_alias

FROM nama\_tabel

Contoh :

SELECT ID as produkID, ProductName AS produk

FROM Produks;

SELECT p. ID FROM produks AS p WHERE p. ID = 2 ;

1. SQL Joins

SQL Joins digunakan untuk menggabungkan baris dari dua atau lebih tabel, berdasarkan kolom terkait diantara mereka.

Contoh :

SELECT Orders, OrderID,

Customers, customersName,

Orders. OrderDate

FROM Orders

INNER Join customers ON

Orders, CustomerID = Customers. CustomerID ;

1. SQL Inner Join

SQL Inner Join digunakan untuk menghasilakan baris data dengan cara menggabungkan 2 buah tabel atau lebih menggunakan pasangan data yang match pada masing-masingtabel. Perintah ini sama dengan join yang sering di gunakan.

Contoh :

SELECT tabeldata.nama.tabeldata.email,orde.no\_order

FROM tabeldata

INNER JOIN order

ON tabeldata id=order.id

ORDER BY tabeldata.nama

1. SQL Left Join

SQL Left Join digunakan untuk menghasilkan baris data dari tabel kiri (nama tabel pertama) yang ada pasangan datanya pada tabel kanan (nama tabel kedua).

Syntax dasar :

SELECT column\_name(s)

FROM table\_name1

LEFT JOIN table\_name2

ON table\_name1.column\_name=table\_name2.

column\_name

contoh :

SELECT tabeldata.nama,tabeldata.email,order.no\_order

FROM tabeldata

LEFT JOIN order

ON tabeldata.id=order.id

ORDER BY tabeldata.nama

1. SQL Right Join

SQL Right Join Digunakan untuk menghasilkan baris data dari tabel kanan (nama tabel kedua) yang tidak ada pasangan datanya pada tabel kiri (nama tabel pertama).

Syntax dasar :

SELECT column\_name(s)

FROM table\_name1

RIGHT JOIN table\_name2

ON table\_name1.column\_name=table\_name2

column\_name

contoh :

SELECT tabeldata.nama,tabeldata.emailmorder.no\_order

FROM tabeldata

RIGHT JOIN order

ON tabeldata.id=order.i

ORDER BY tabeldata.nama

1. SQL Full Join

Digunakan untuk menghasilkan baris data jika ada data yang sama pada salah satu tabel.

Syntax dasar :

SELECT column\_name(s)

FROM table\_name1

FULL JOIN table\_name2

ON table\_name1.column\_name=table\_name2

column\_name

Contoh :

SELECT tabeldata.nama,tabeldata.email,order.no\_order

FROM tabeldata

FULL JOIN order

ON tabeldata.id=order.id

ORDER BY tabeldata.nama

1. SQL Selft Join

SQL Selft Join adalah join biasa tetapi tabel bergabung dengan dirinya sendiri.

Syntax :

SELECT column\_name(s)

FROM tabel TI, table T2

WHERE condition;

Contoh :

SELECT A CustomerName AS

CustomerName1, B.CustomerName AS

CustomerName2, A,city

FROM Customer A, customerID<> B. CustomerID

AND A City = b.City

ORDER BY A.City;

1. SQL Union

Digunakan untuk menggabungkan hasil dari 2 atau lebih perintah Select.

Syntax dasar :

SELECT column\_name(s)FROM table\_name1

UNION column\_name(s) FROM table\_name2

Atau

SELECT column\_name(s) FROM table\_name1

UNION ALL

SELECT column\_name(s) FROM table\_name2

Contoh :

SELECT nama FROM mhs\_kampus1

UNION

SELECT nama FROM mhs\_kampus2

1. SQL Group By

Digunakan untuk mengelompokkan baris yang memiliki nilai yang sama ke dalam baris rin gkasan.

Pernyataan Group By sering digunakan dengan fungsi agregat (COUNT(), MAX(), MIN(), SUM(), AVG()) untuk mengelompokkkan kumpulan hasil menurut satu atau beberapa kolom.

Syntax:

SELECT column\_name (s)

FROM table\_name

WHERE condition

GROUO BY column\_name(s)

ORDER BY column\_name (s);

Contoh:

SELECT shippers.shipperName,

COUNT (order . orderID) AS

Numberoforders FROM orders

LEFT JOIN Shippers ON

Orders.ShippersID =

Shippers.ShipperID

GROUP BY ShipperName;

Contoh:

SELECTCOUNT(CostumerID), country

FROM costumers

GROUP BY country

ORDER BY COUNT (costumerID) DESC;

1. SQL Having

Pada Clause HAVING itu digunakan pada pernyataan SELECT guna untuk menentukan kondisi filter untuk sekelompok baris atau pun agrerat.

**Contoh:**

Pada pernyataan ini adalah pernyataan yang menentukan jumlah pada jumlah pesanan barang yang dihitung pada jumlah ID toko tersebut.

SELECT (ShopID),Price

FROM ShopID

WHERE Condition

GROUP BY Price

WHERE Condition

HAVING COUNT (ShopID ) > 4;

1. SQL Exists

EXISTS itu untuk mengembalikan sebuah nilai true

Pada operator Exists itu digunakan untuk menguji rekaman yang ada pada di subquery.

Contoh:

SELECT Buyer’sName

FROM Buyer

WHERE EXISTS (SELECT NameOfGoods FROM Goods WHERE Goods.Buyer = Buyer’s.Buyer’s AND Price < 50);

1. SQL Any, All

ANY dan ALL dalam kata kunci itu digunakan dengan WHERE atau bisa juga dengan HAVING, Yang pada kedua operasi itu mereka beroperasi pada subquery yang dapat mengembalikan banyak nilai.

Contoh:

SELECT NameOfGoods

FROM Goods

WHERE GoodsID operator ANY

(SELECT GoodsID FROM Orders WHERE Total = 5);

SELECT NameOfGoods

FROM Goods

WHERE GoodsID operator ALL

SELECT GoodsID FROM Orders WHERE Total = (5);

1. SQL SELECT INTO

SELECT INTO dapat menyalin data dari satu tabel ke tabel yang baru. Pada operasi ini kita membuat tabel baru pada grub file default.

Contoh:

* Pada pernyataan SQL berikut ini adalah untuk membuat cadangan untuk para pembeli:

SELECT \* INTO BuyersBackup(2020)

FROM Buyers;

* Pada pernyataan yang berikut ini adalah menggunakan clause IN yang untuk digunakan untuk menyalin dari tabel ke tabel yang baru di data base lainnya.

SELECT \* INTO BuyersBackup(2020) IN ‘Backup.mdb’

FROM Buyers;

* Pada pernyataan berikut ini adalah hanya dengan menyalin beberapa kolom ke dalam tabel yang baru:

SELECT NameOfGoods,ItemCode INTO BuyersBackup(2020)

FROM Buyers;

* Pada pernyataan yang berikut ini adalah untuk menyalin data dari satu tabael ke dalam tabel yang baru:

SELECT ShopID FROM ItemCode(Tabel 1)

UNION

SELECT ShopID FROM NameOf Goods(Tabel 2);

1. SQL INSERT INTO SELECT

Pernyataan INSERT INTO SELECT dapat menyalin data dari satu tabel dan menyimpannya ke tabel yang lain.

Contoh:

Untuk pernyataan berikut ini adalah menyalin kedua tabel yang dimana “penjual” ke “pembeli” telah kita buat sebelumnya (isi semua kolom):

INSERT INTO (ShopID, Address, ShopName, TransactionNumber, Period, ItemCode, TheAmountOfGoods, NameOfGoods, Price)

SELECT Buyer’sName, Address, ShopName, TransactionNumber, Period, ItemCode, TheAmountOfGoods, NameOfGoods, Price FROM Buyer’sName;

1. SQL CASE

Pernyataan CASE melewati kondisi dan mengembalikan nilai pada saat kondisi pertama yang terpenuhi (seperti pada pernyataan IF-THEN-ELSE).

Contoh:

SELECT TransactionName,Price,

CASE

WHEN Price > 30.000 THEN ‘The price is greater than 30.000’

WHEN Price = 30.000 THEN ‘The price is 30.000’

ELSE ‘The price is under 30.000’

END AS Price

FROM ShopID;

1. SQL Null Values

Bidang dengan nilai NULL adalah bidang tanpa nilai. Jika bidang dalam tabel adalah opsional, dimungkinkan untuk menyisipkan catatan baru atau memperbarui catatan tanpa menambahkan nilai ke bidang ini. Kemudian, bidang tersebut akan disimpan dengan nilai NULL.

Contoh:

SELECT ItemName,TotalPrice \* (TheNumberOfGoods + OrderQuantity)

FROM Item;

1. SQL Stored Procedures

Stored Procedures adalah kode SQL yang disiapkan yang dapat Anda simpan, sehingga kode tersebut dapat digunakan kembali berulang kali.

Contoh:

CREATE PROCEDURE selectAllBuyer’s

AS

SELECT \* FROM Buyer

GO;

1. SQL Comments

Comments digunakan untuk menjelaskan bagian dari pernyataan SQL, atau untuk mencegah eksekusi pernyataan SQL.

Contoh:

Berikut ini adalah contoh yang menggunakan komentar multi-baris sebagai penjelasan berikut dibawah ini:

/\* select all the columns

Of all the records

In the Buyer table :\*/

SELECT \* FROM Buyer;

1. SQL Operators

Kita ketahui bahwa yang dimaksud operator dalam SQL adalah kata atau karakter yang khusus yang sering digunakan terutama dalam clause WHERE pada pernyataan SQL untuk melakukan operasi, misalnya seperti pada perbandingan dan operasi aritmatika.

Contoh Deskripsi Operators

Tambahkan (+), Kurangi(–), Kalikan (\*), Membagi (/), Modul (%)

**B. SQL DATABASE**

1. SQL Database

Apa yang dimaksud dengan database? Database adalah tempat menyimpan data.

1. SQL Drop DB

Pernyataan DROP DATABASE digunakan untuk menjatuhkan database SQL yang ada.

1. SQL Backup DB

Pernyataan BACKUP DATABASE digunakan di SQL Server untuk membuat cadangan penuh dari database SQL yang ada.

1. SQL Create Table

Pernyataan CREATE TABLE digunakan untuk membuat tabel baru dalam database.

1. SQL Drop Table

Pernyataan DROP TABLE digunakan untuk menjatuhkan tabel yang ada dalam database.

1. SQL Alter Table

Pernyataan ALTER TABLE digunakan untuk menambah, menghapus, atau memodifikasi kolom dalam tabel yang sudah ada. Pernyataan ALTER TABLE juga digunakan untuk menambah dan menghapus berbagai batasan pada tabel yang ada.

1. SQL Constraints

Constraint digunakan untuk membatasi jenis data yang dapat masuk ke dalam tabel. Ini memastikan keakuratan dan keandalan data dalam tabel. Jika ada pelanggaran antara batasan dan tindakan data, tindakan dibatalkan.

Contoh:

CREATE TABLE table\_name (

ShopID datatype (constraint),

NameOfGoods datatype (constraint),

Price datatype (constraint),

);

1. SQL not NULL

SQL Not Null yaitu bidang tanpa nilai. Jika bidang dalam tabel bersifat opsional, dimungkinkan untuk menyisipkan catatan baru atau memperbarui catatan tanpa menambahkan nilai ke bidang ini. Kemudian, field akan disimpan dengan nilai NULL.

Contoh:

CREATE TABLE GoodsShop (

ShopID int NOT NULL,

NameOfGoods Varchar(255) NOT NULL,

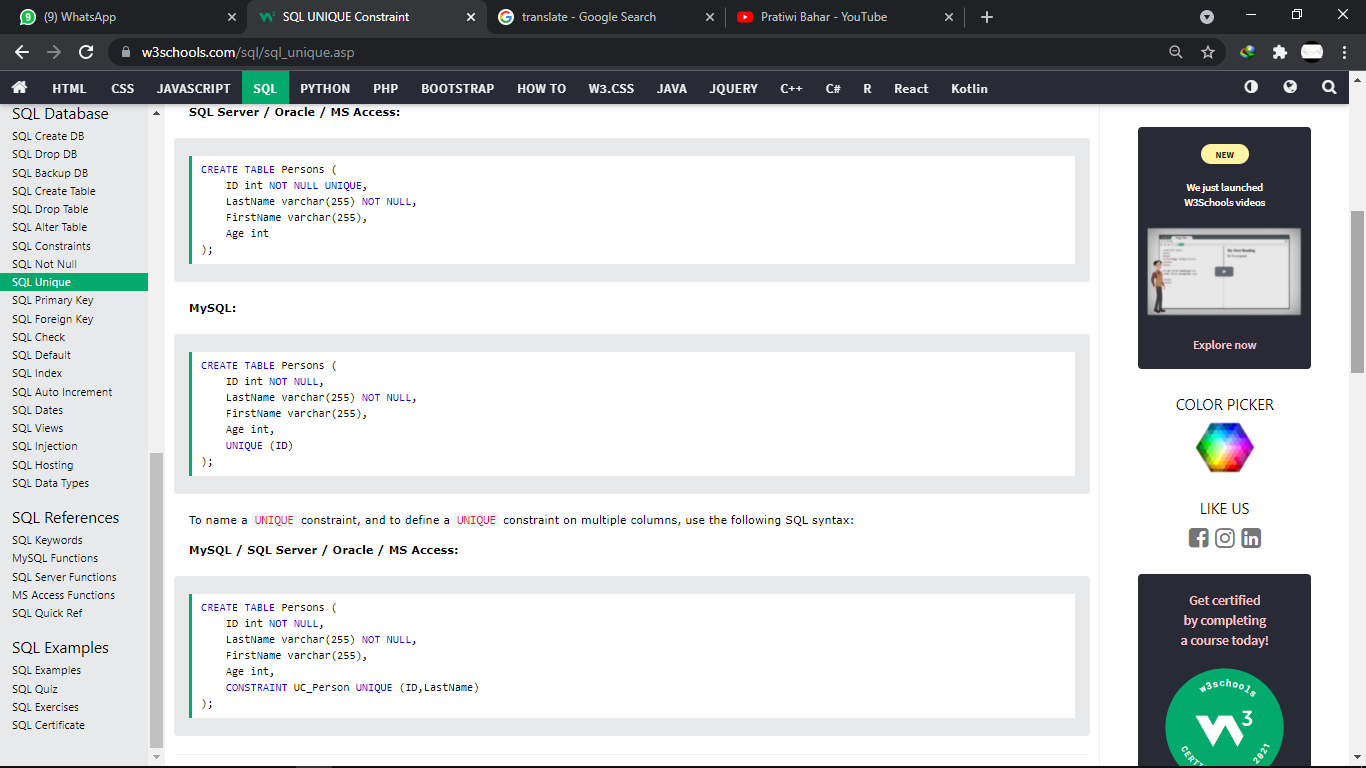
Price Varchar(255) NOT NULL,

Total int

);

1. SQL Unique yaitu, Batasan UNIQUE memastikan bahwa semua nilai dalam kolom berbeda.

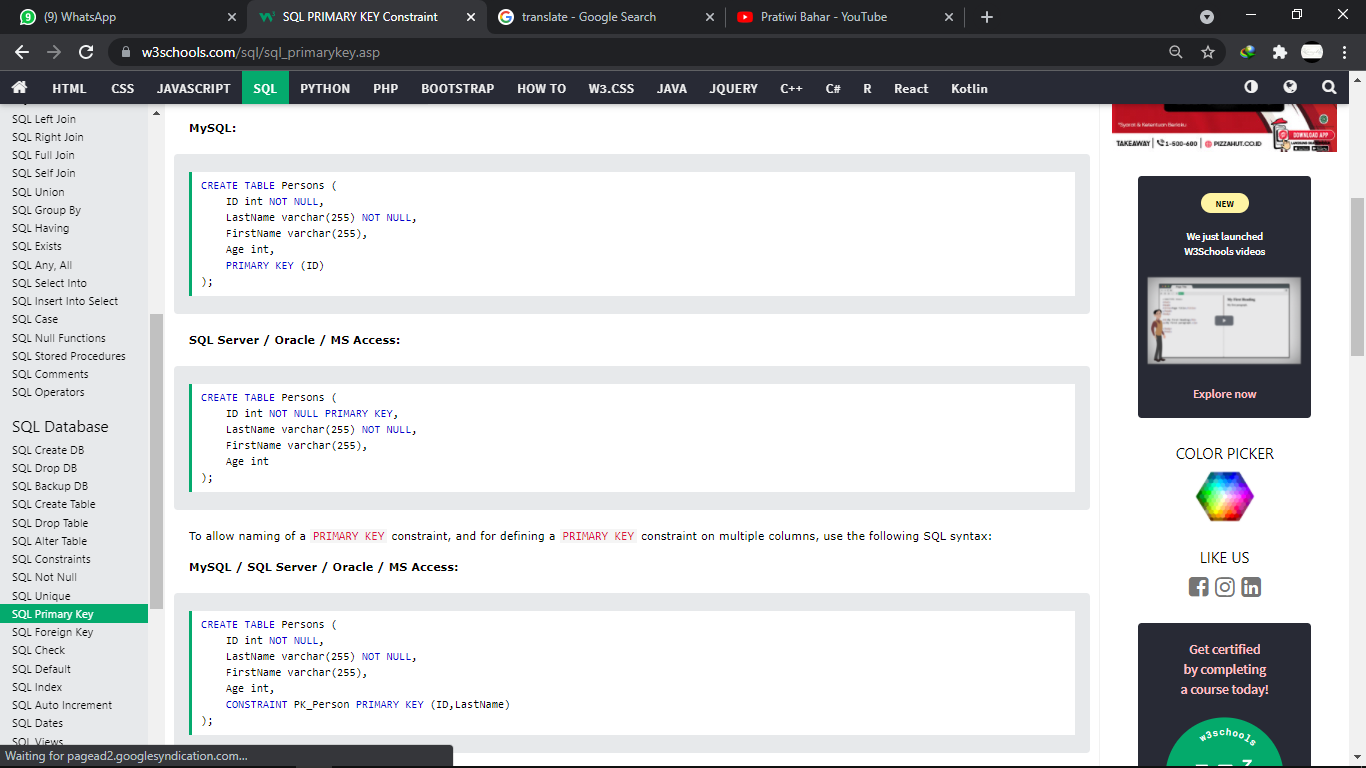
Contoh



1. SQL PRIMARY KEY

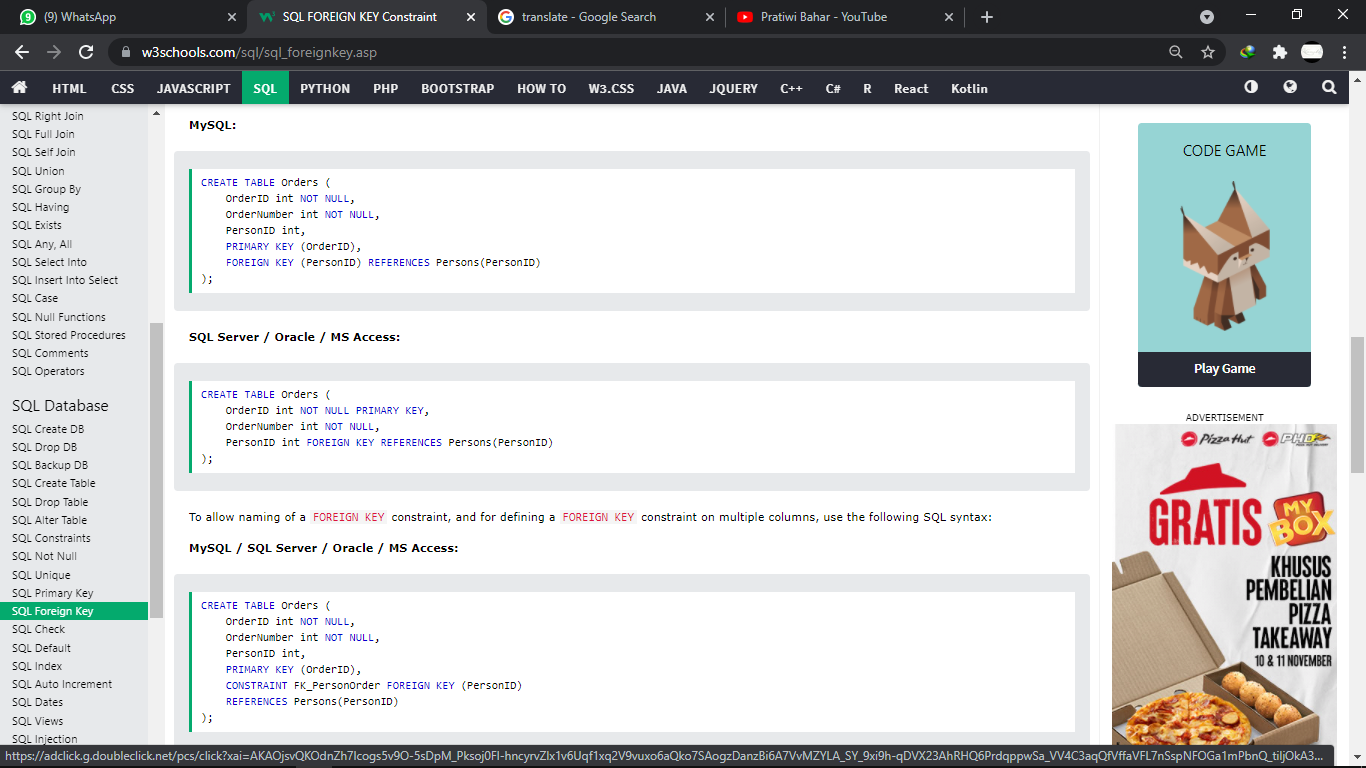
Batasan primary key secara unik mengidentifikasi setiap record dalam table.

Contoh:



1. SQL Foreign Key Batasan FOREIGN KEY digunakan untuk mencegah tindakan yang akan menghancurkan link antar tabel. KUNCI ASING adalah bidang (atau kumpulan bidang) dalam satu tabel, yang merujuk ke KUNCI UTAMA di tabel lain.

Contoh



1. SQL Check

Batasan CHECK digunakan untuk membatasi rentang nilai yang dapat ditempatkan dalam kolom. Jika Anda menentukan batasan CHECK pada kolom, itu hanya akan mengizinkan nilai tertentu untuk kolom ini.

Contoh:

CREATE TABLE GoodsShop (

ShopID int NOT NULL,

NameOfGoods Varchar(255) NOT NULL,

Price Varchar(255),

Total int,

CHECK (Total>=18)

Akses SQL Server / Oracle / MS:

CREATE TABLE GoodsShop (

ShopID int NOT NULL,

NameOfGoods Varchar(255) NOT NULL,

Price Varchar(255),

Total int, CHECK (Total>=18)

);

Pada penamaan batasan PERIKSA, dan untuk batasan PERIKSA, pada beberapa kolom, dan menggunakan sintaks SQL berikut ini.

CREATE TABLE GoodsShop (

ShopID int NOT NULL,

NameOfGoods Varchar(255) NOT NULL,

Price Varchar(255),

Total int,

Address Varchar(255),

CONSTRAINT CHK\_GoodsShop CHECK (Total>=18 AND = ‘jln.cendrawesi 200’)

);

1. SQL default

Digunakan untuk memberikan nilai default untuyk sebuah kolom.

Contoh:

CREATE TABLE GoodsShop (

ShopID int NOT NULL,

NameOfGoods Varchar(255) NOT NULL,

Price Varchar(255),

Total int,

Address Varchar(255)DEFAULT ‘jln.cendrawesi 200’

);

Batasan DEFAULT itu biasanya dapat juga kita gunakan untuk memasukkan nilai sistem, dengan menggunakan fungsi sepsrti GETDATE ():

CREATE TABLE Buyer(

BuyerID int NOT NULL,

BuyerNumber int NOT NULL,

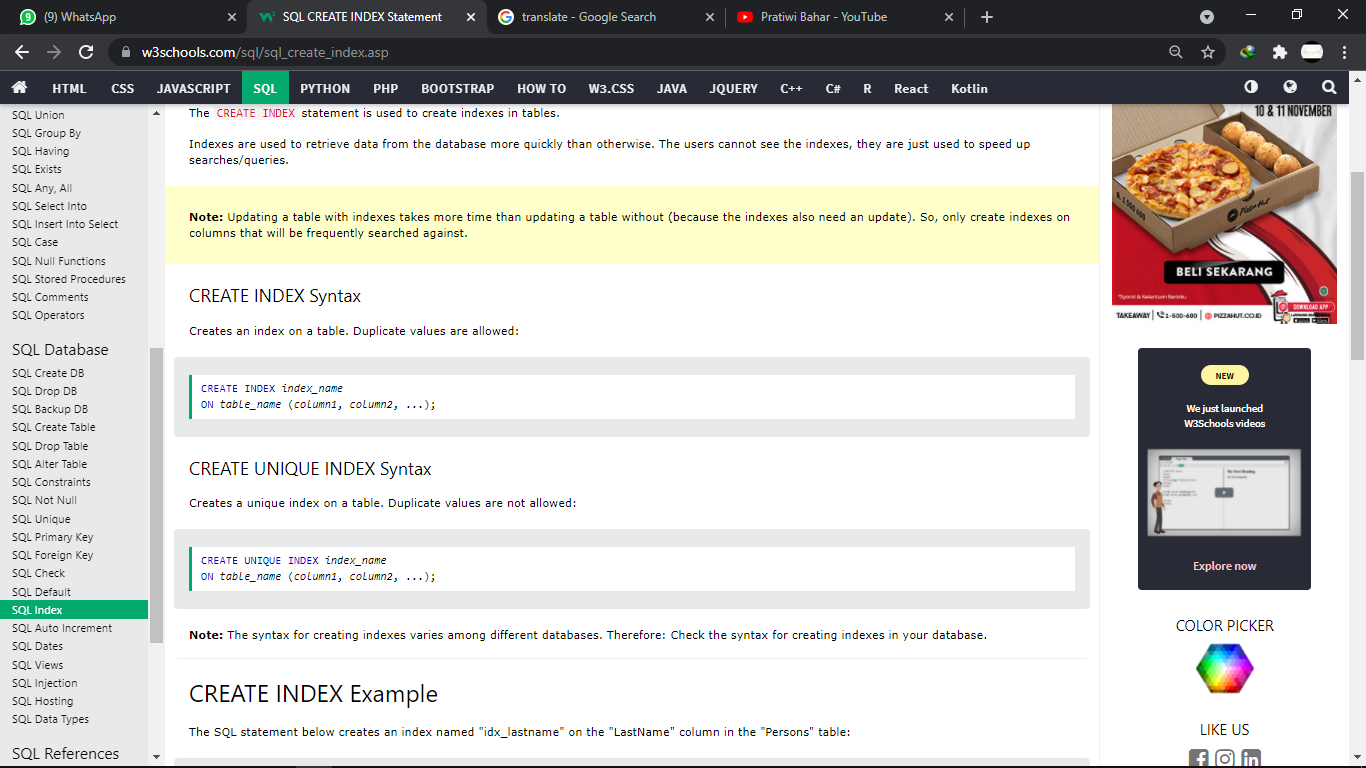
BuyerDate date DEFAULT GETDATE ()

);

1. SQL INDEX

SQL Index adalah sebuah objek dalam sistem database yang dapat mempercepat proses pencarian (query) data.

Contoh



1. SQL Auto Increment adalah Peningkatan otomatis memungkinkan nomor unik dihasilkan secara otomatis ketika catatan baru dimasukkan ke dalam tabel.

Contoh:

CREATE TABLE GoodsShop (

ShopID int NOT NULL, AUTO\_INCREMENT,

NameOfGoods Varchar(255) NOT NULL,

Price Varchar(255),

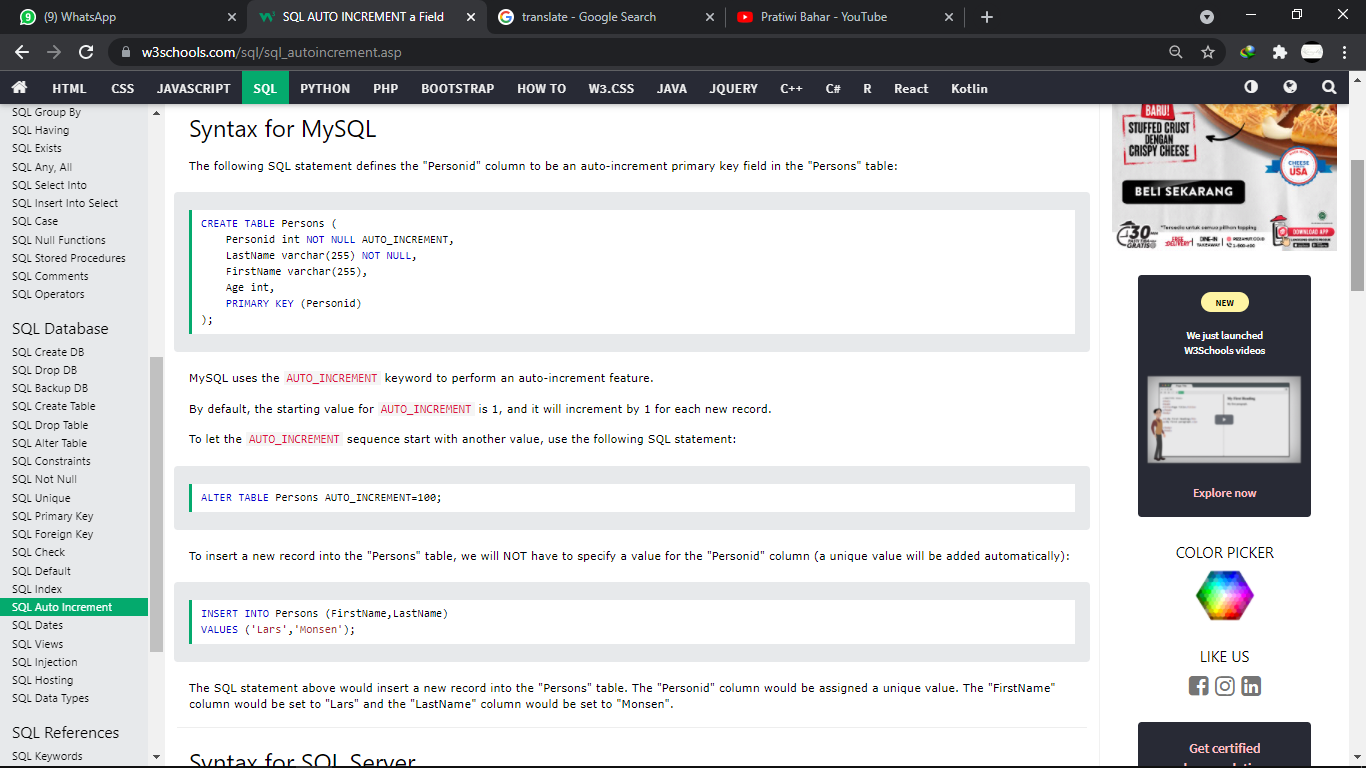
Total int,

PRIMARY KEY (Personid)

);

1. SQL Dates merupakan Bagian tersulit saat bekerja dengan tanggal adalah memastikan bahwa format tanggal yang Anda coba masukkan, cocok dengan format kolom tanggal dalam database.

Contoh:



1. SQL Dates

Bagian tersulit saat bekerja dengan tanggal adalah memastikan bahwa format tanggal yang Anda coba masukkan, cocok dengan format kolom tanggal dalam database.

Contoh:

SELECT \* FROM Buyer WHERE BuyerDate=’2020-12-25’

1. SQL VIEWS

SQL VIEWS adalah perintah query yang disimpan pada database dengan suatu nama tertentu, sehingga bisa digunakan setiap saat untuk melihat data tanpa menuliskan ulang query tersebut dan dapat didefenisikan sebagai **‘tabel virtual’**.

Contoh:

CREATE VIEW Syntax

CREATE VIEW view\_name AS

SELECT COLUMN1, COLUMN2...

FROM Table\_name

WHERE condition;

Pada SQL berikut ini membuat tampilan yang memperlihatkan seluruh buyers dari Indonesia

CREATE VIEW (Indonesia Buyer) AS

SELECT Buyer’sName, ItemCode

FROM Buyer

WHERE Country = ‘indonesia’;

Pada SQL berikut ini adalah untuk membuat tampilan yang memilih pada setiap barang dalam tabel “Goods” dengan harga yang besar dari harga yang rata-rata:

CREATE VIEW (Goods Above Average Price)AS

SELECT NameOfGoods, Price

FROM Goods

WHERE Price > (SELECT AVG(Price) FROM Goods);

Berikut ini kita dapat query view di atas sebagai berikut dibawah ini:

SELECT \* FROM (Goods Above Average Price

1. SQL INJECTION

adalah teknik injeksi kode yang dapat menghancurkan database Anda. SQL Injection adalah salah satu teknik peretasan web yang paling umum. SQL Injection adalah penempatan kode berbahaya dalam pernyataan SQL, melalui input halaman web.

Contoh:

txtUserId = getRequestString(“UserId);

txtSQL = “SELECT \* FROM User WHERE UserId = “ + txtUserId;

SQL Injectoion berdasarkan 1=1 selalu benar

SELECT \* FROM User WHERE UserId = 110 OR 1=1;

1. SQL HISTONG

Jika Anda ingin situs web Anda dapat menyimpan dan mengambil data dari database, server web Anda harus memiliki akses ke sistem database yang menggunakan bahasa SQL. Jika server web Anda di-host oleh Penyedia Layanan Internet (ISP), Anda harus mencari paket hosting SQL. Database hosting SQL yang paling umum adalah MS SQL Server, Oracle, MySQL, dan MS Access.

1. SQL Data Types

Setiap kolom dalam tabel database harus memiliki nama dan tipe data. Pengembang SQL harus memutuskan jenis data apa yang akan disimpan di dalam setiap kolom saat membuat tabel. Tipe data adalah pedoman bagi SQL untuk memahami tipe data apa yang diharapkan di dalam setiap kolom, dan juga mengidentifikasi bagaimana SQL akan berinteraksi dengan data yang disimpan.

1. SQL Data Types

Tipe data kolom menentukan nilai apa yang dapat disimpan kolom: bilangan bulat, karakter, uang, tanggal dan waktu, biner, dan seterusnya.

1. **SQL References**
2. SQL Keywords

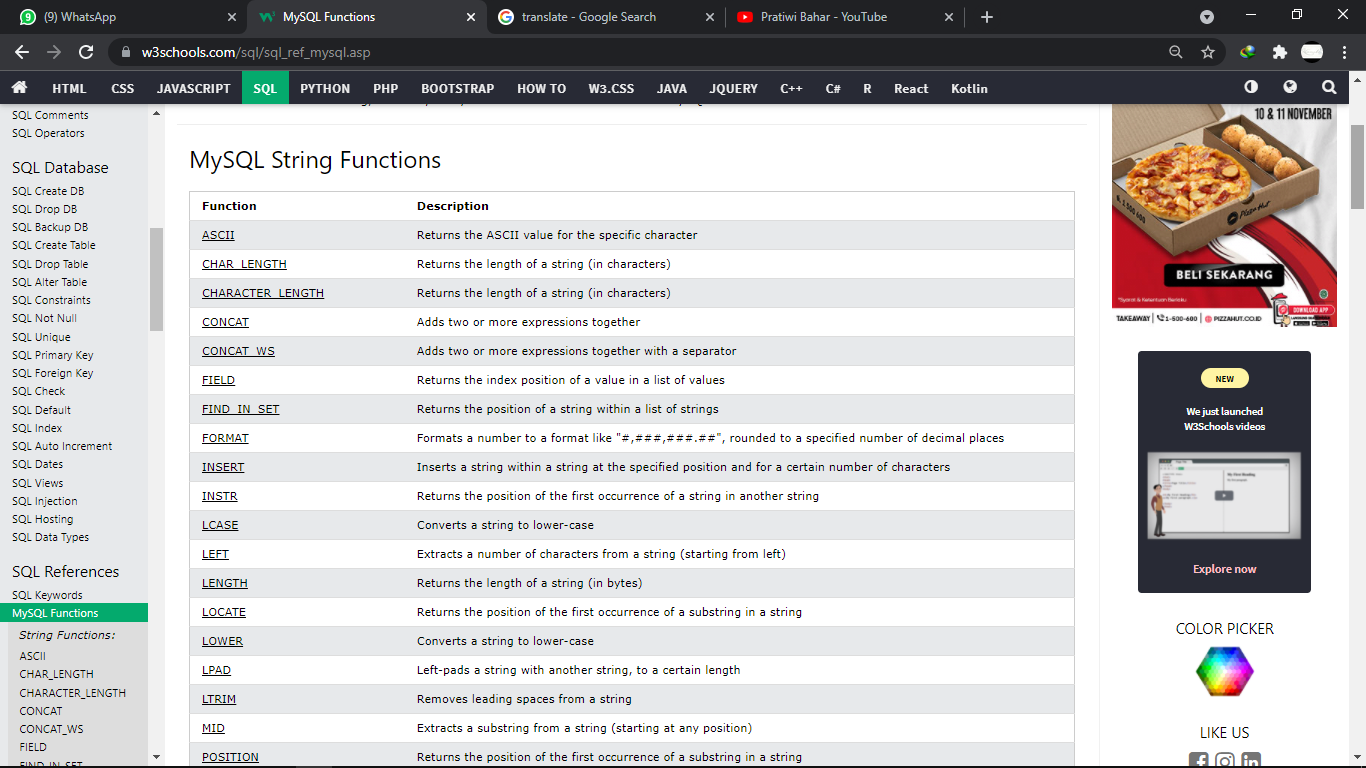
Referensi kata kunci SQL ini berisi kata-kata yang dicadangkan dalam SQL.

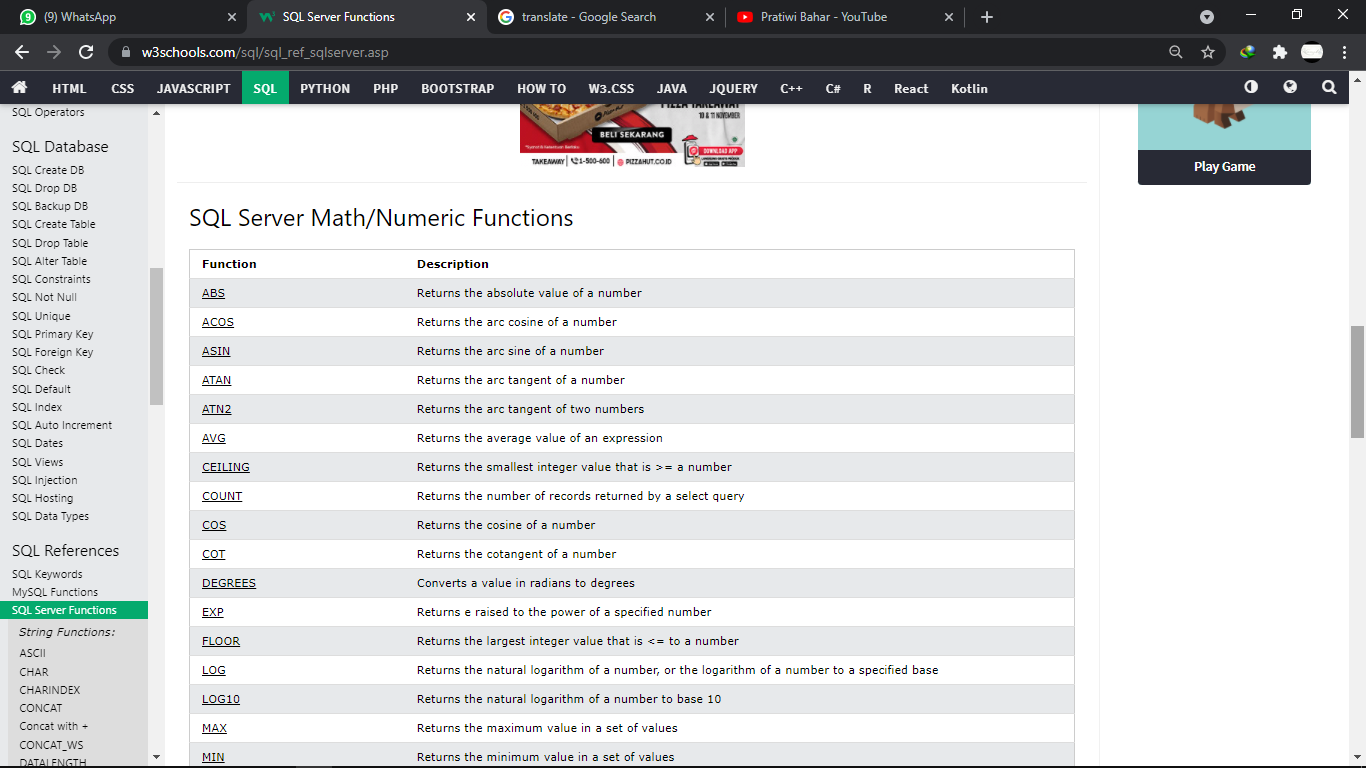
1. MySQL Functions

Referensi ini berisi string, numerik, tanggal, dan beberapa fungsi lanjutan di MySQL.

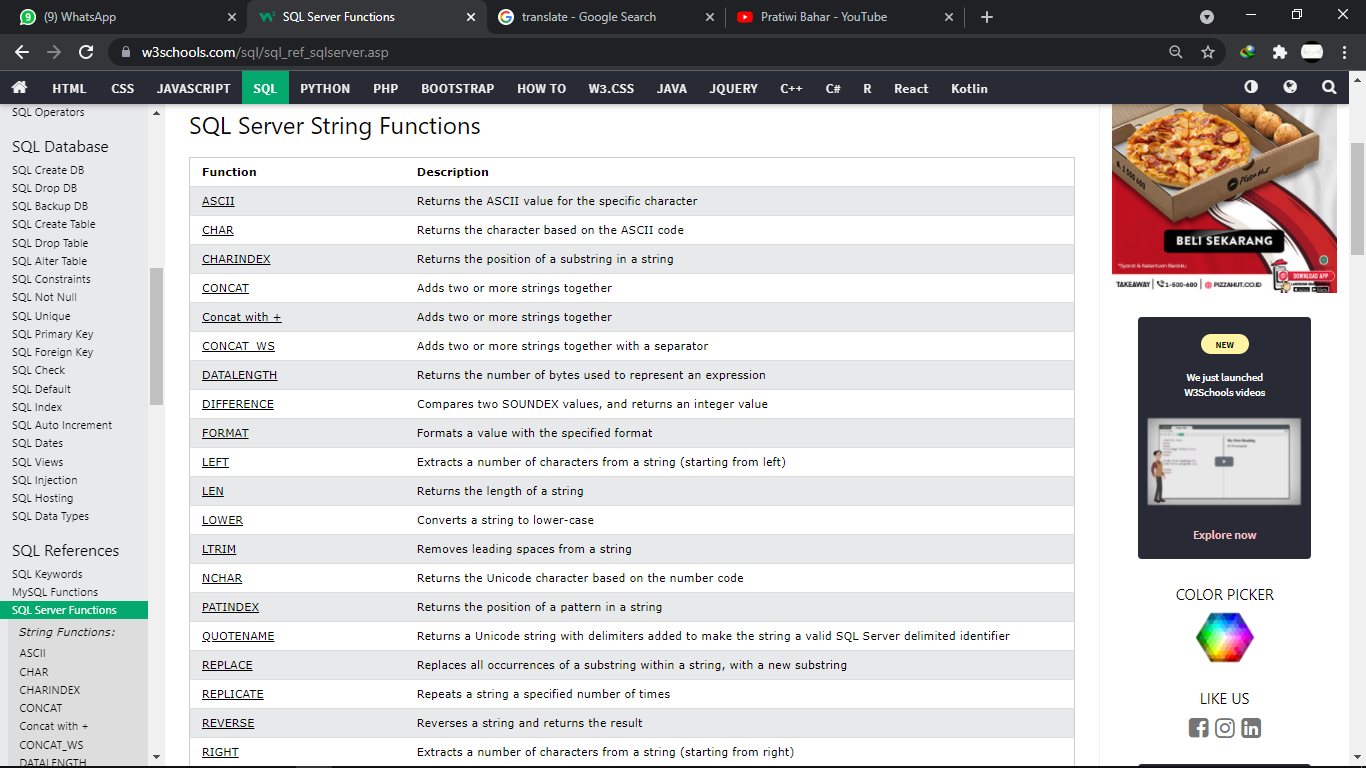
1. MySQL Functions Referensi ini berisi string, numerik, tanggal, dan beberapa fungsi lanjutan di MySQL.

Contoh





1. SQL Server Functions SQL Server memiliki banyak fungsi bawaan. Referensi ini berisi string, numerik, tanggal, konversi, dan beberapa fungsi lanjutan di SQL Server.

Contoh:

1. MS Access Functions

Referensi ini berisi fungsi string, numerik, dan tanggal di MS Access.

1. SQL Quick Ref

|  |  |
| --- | --- |
| SQL Statement | Syntax |
| AND / OR | SELECT column\_name(s) FROM table\_name WHERE condition AND|OR condition |
| ALTER TABLE | ALTER TABLE table\_name ADD column\_name datatype  or  ALTER TABLE table\_name DROP COLUMN column\_name |