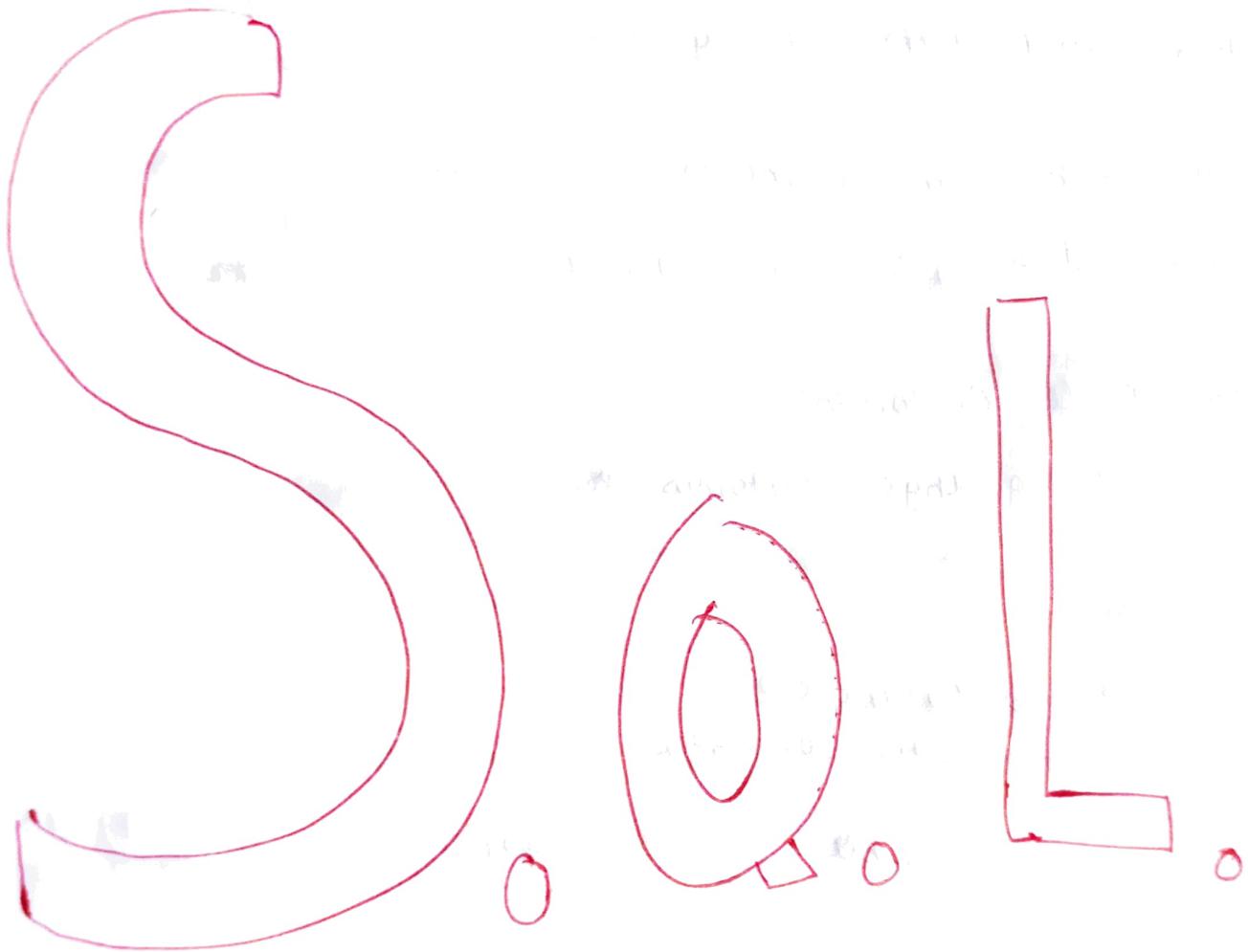
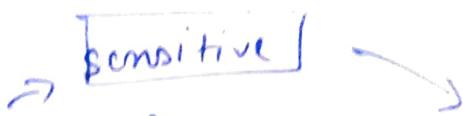


S.O.L



By - colt - steel
=

SQL statements



your database

Select * FROM Customers;

Capital / Small NO ~~mat~~

→ Select * FROM Products ORDER BY Price DESC;

Nothing except Table-name is case sensitive

DESC , desc , order by Order By ... etc.

Table name	Records
Customers	91
Categories	8
Employees	10
Order Details	518
Orders	196
Products	77
Shipments	3
Suppliers	29

DROP Table Customers;

↳ goodbye customers 

Section-2

what is a database?

a collection of data

A quick example

Andrea, Archie - (949) 345-2222

phonebook 

::

Stank, Ned .

(310)-119-6501

Q. Find people with first name 'Archie'?

or
arvy



Q. find all phone numbers of area code 949

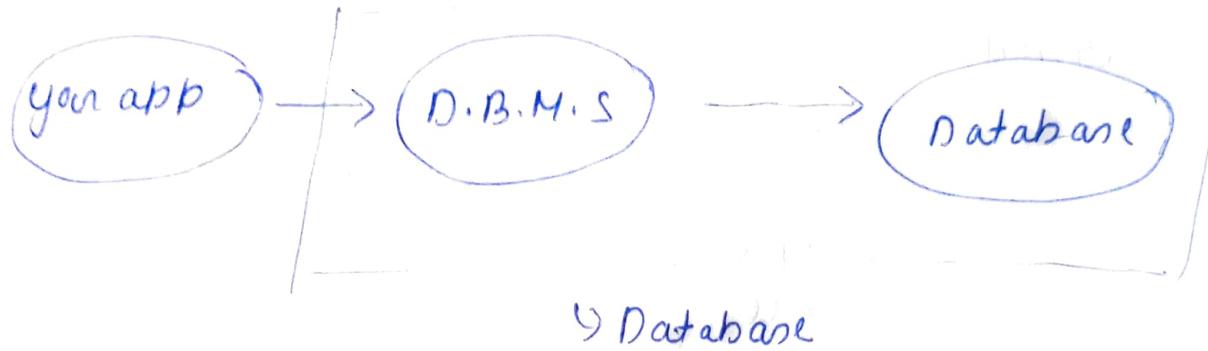
Q. find all the people who have the three letter
first name?

Q. what is a database?

→ ① a collection of data

→ ② a method of accessing and manipulating data

Database vs Database Management System



For managing this database PostgreSQL MySQL

Oracle Database SQLite

Finally? what is a database

A structured set of computerized data with an accessible interface

Data is knowledge and money



Some popular databases

MongoDB, Oracle, MySQL, SQLite, PostgreSQL, MaxDB, Firebird, Redis

MySQL vs SQL

SQL → it is the language that we use to talk to our users

ex. find all users, find all users who are 18 years old, Add a new user with a username, Jamborim, Delete every single user... etc

A quick preview

Find all users who are 18 or older

Select * From users WHERE Age >= 18;

Relational Database

MYSQL

SOLITE

PostgreSQL

Oracle



→ All uses SQL

Syntax difference

MySQL

Select * FROM users WHERE Age >= 18;

PostgreSQL

Select * FROM users WHERE Age >= 18;

Almost same 😊 but not always

Two takeaways

- ① Once we learn SQL it is pretty easy to use or switch to another DB that uses SQL
- ② What makes the (DBMS) unique is the features they offer not the language.

Some code

Select * from animals;

Select species, status FROM animals;

Section-3

→ mysql -u root -p

Clean the screen

ctrl + l

~~Command~~ Show databases;

→ Creating databases

Command → CREATE DATABASE < database-name >

Ex. create database chicken-coop;

PetShop

✗ shivam "kumar";
No spacing

→ Dropping database

Command → DROP DATABASE chicken-coop;

→ To use a database

USE < database-name >;

| Select database(); |

Database types: The basics

VARCHAR → variable char.

↳ Must be text

Ex. VARCHAR(100) ⚡ limit set on ~~to~~ ^{at} 100 words or
↑ specifying the size

INTEGER → INT

Q. Draw a tweet table

Username	Content	Favorites
VARCHAR(15)	VARCHAR(140)	INT
'coolguy'	'my-first-tweet!'	1
'guitar-queen'	'I love music :)'	10

→ Creating tables ←

Syntax → CREATE TABLE <tablename>

```
( column_name data-type )
    column_name data-type
)
;
```

Ex. create table CATS

```
( name VARCHAR(100),
    age INT
)
;
```

Q. Creating Table

- ① use chicken_coop;
- ② select database();
- ③ create table chicken(name VARCHAR(50), age INT);

Q. How to check the table created worked or not?

use → command → show Tables;

Q. How to know about columns

Command → show COLUMNS From ~~chicken~~ chicken;

Field	Type	Null	Key	Default	Extra
name	varchar(50)	yes		NULL	
age	int	yes		NULL	

or

DESC <tablename>;

describe ~~chicken~~ chicken;

→ Deleting Tables ←

Syntax → DROP TABLE <tablename>;

Ex. drop table chicken;

→ Comments in SQL ←

-- I am commenting something

spare
(important)

6 Section-4

6 gnsnt - The basics

Adding data to your table

Syntax `gnsnt gnto tablename (name, age)`
`values ('Jetson', 7);`

Q. How do we know it worked?

Command → Select * from cats;

→ Multiple gnsnts ←

command → gnsnt gnto cats (name, age)
`values ('charlie', 10), ('sadie', 3), ('Lazy Bear', 1);`

→ Working with not null ←

→ due people;

Field	Type	Null	Key	Default	Extra
first_name	varchar(30)	yes		Null	
second_name	varchar(30)	yes		Null	
age	int	yes		Null	

Null → The value is not known, Null doesn't mean 0

Command → gnsnt gnto cats(name) value ('Todd');

We didn't provide age this time still it worked



Name	age
Brath	5
Meatball	7
Todd	Null

command → insert into cats() values();

Name	age
..	..
Null	Null

Q- what if u don't want your table to accept Not null values?

→ NOT NULL ←

Syntax,

```
create Table cats 2
( name varchar(100) NOT NULL ,
  age int NOT NULL
);
```

Example → create table cats 2 (name) values ("charlie");

Error → Field 'age' doesn't have a default value;

Now we cannot leave the age field empty 😠

Use cats;

Field	Type	Null	Key	Default	Extra
..

Drop cats 2;

Field	Type	Null	Key	Default	Extra
..

GF name is like 'mario's pizza'

use

'mario's pizza'

↑ skipping quotation

or

'mario said "Haha!"'

↳ is also ok!

→ 'Adding default values' ↵

Field name	Type	Null	key	Default	Extra
name	varchar(50)	yes		Null	
age	int	yes		Null	

→ To set default values ↵

Syntax → CREATE TABLE cats3

```
( name varchar(100) Default 'unnamed',  
    age INT Default 99  
);
```

Create Table cats4

```
( name varchar(100) not null Default 'unnamed'  
    age int not null Default 99  
);
```

GF name is like 'mario's pizza'

use 'mario's pizza'

↑ skipping quotation

or [mario said "Maha"]

↳ is also ok!

→ ' Adding default values' ←

Field name	Type	Null	key	Default	Extra
name	varchar(50)	yes		NULL	
age	int	yes		NULL	

→ To set default values ←

Syntax → CREATE TABLE cats3

(name varchar(100) Default 'unnamed',
age int Default 99
);

create Table cats4

(name varchar(100) not null Default 'unnamed'
age int not null default 99
);

GF name is like "mario's pizza"

use

"mario's pizza"

↑ skipping quotation

or

'mario said "Maha!"'

↳ is also ok!

→ "Adding default values" ↵

Field name	Type varchar(50)	Null yes	key	Default Null	Extra Null
age	int	yes			

→ To set default values ↵

Syntax → CREATE TABLE cats3

(name varchar(100) Default 'unnamed',
age int Default 99
);

create Table cats4

(name varchar(100) not null Default 'unnamed'
age int not null Default 99
);

Insert into Cats3(name, age) values(?, ?);

Select * From Cats3;

name	age
mystery	2
mystery	99
mystery	99

→ we are getting some default values

→ if the default is not null we can set the value to be null manually

like this.

Grant into Cats3(name, age)
values (NULL, 3);

Create table Cats3;

? ..

grant into Cats3 (age) values ('3');

Select * From Cats3;

name	age
unnamed	13

all empty

grant into Cats3() values ();

name	age
unnamed	13
unnamed	99

Insert into Cats3(name, age) values();

Select * from Cats3;

name	age
mystery	2
mystery	99
mystery	99

→ we are getting some default values

→ if the default is not null we can set the value to be null manually

like this.

Insert into Cats3(name, age)

values (NULL, 3);

Create table Cats3;

7 ..

grant insert into Cats3 (age) values ('3');

Select * from Cats3;

name	age
unnamed	13

all empty

grant insert into Cats3() values ();

name	age
unnamed	13
unnamed	99

→ Combine not null and default ←

Cats-4

==

→ insert into ~~cats~~ cats4 () values ();
Select * from cats4 ;

name	age
unnamed	99

→ insert into cats4 (name) values ('tommy');

name	age
unnamed	99
tommy	99

→ introduction to primary keys ←

Field	Type	Null	Default	Extra
name	VARCHAR(100)	NO	unnamed	..
age	int	NO	99	..

Primary key
(unique identification)

Command →

CREATE TABLE unique_cats (

cat_id INT PRIMARY KEY,
space

name VARCHAR(100) NOT NULL,
age INT NOT NULL

);

Another option

```
CREATE TABLE unique-cats(2) (
    cat-id INT, name VARCHAR(100) NOT NULL,
    age INT NOT NULL, PRIMARY KEY (cat-id)
);
```

command → `insert into unique-cats(cat-id, name, age)`
values (1, 'muchikin', 23)

cat-id	name	age
1	muchikin	23

previous

if i repeat command with the same cat-id → 1 . it will
show → Duplicate entry '1' for keys. 'unique-cats.PRIMARY'

→ repeat the previous command by changing only cat-id
gives

catid	name	age
-999	muchikin	23
1	muchikin	23
2	muchikin	23
3	muchikin	23

→ working with Auto Increment

Insert into unique-cats (cat-id, name, age)
values (X 'ramu', 2);

This time we just don't give cat-id and its value
specified

Output → ERROR 1364 Field cat-id doesn't have a default value;

use cat unique-cats;

Field	Type	Null	Key	Default	Extra
cat-id	int	NO	PRI	NULL	
name	varchar(100)	YES		NULL	
age	int	YES		NULL	

→ PRIMARY KEYS CANNOT BE NULL

```
CREATE TABLE unique-cats 3  
cat-id INT AUTO_INCREMENT,  
name varchar(100),  
age INT,  
primary key (cat-id)  
);
```

cat-id will automatically increment for each new cat inserted into the table

gnsnt into unique-cats3 (name, value age)
value ('Boingo' , 1)

output →

cat-id	name	age
1	Boingo	1

Now we can leave the cat-id field empty 😊

or

create table unique-cats4 (cat-id INT AUTO_INCREMENT
PRIMARY KEY,
name varchar(100),
age INT);

Desc unique-cats4;

Field	Type	Null	Key	Default	Extra
cat-id	int	NO	PRI	NULL	auto_increment
name	varchar(100)	YES		NULL	
age	int	YES		NULL	

gnsnt into unique-cats4 () values ();
..
x 4 times

select * from unique-cats4 ;

cat-id	Name	age
1	NULL	NULL
2	NULL	NULL
3	NULL	NULL

Q. Create table (Grant Excise)

→ CREATE TABLE Employees (→ id INT AUTO_INCREMENT
first-name VARCHAR(100), NOT NULL,
last-name VARCHAR(100) NOT NULL,
middle-name VARCHAR(100),
age INT NOT NULL,
current-status VARCHAR(100) NOT NULL
DEFAULT 'Employed',
PRIMARY KEY (id));

OR

→ id INT AUTO_INCREMENT PRIMARY KEY,

Field	Type	Null	Key	Default	Extra
id	int	NO	PRI	NULL	auto-increment
last-name	varchar(100)	NO		NULL	" "
first-name	varchar(100)	NO		" "	" "
middle-name	varchar(100)	YES		" "	" "
age	INT	NO			
current-status	varchar(100)	YES NO		EMPLOYED	

Section 5

(CRUD BASICS)

→ Create, Read, Update, Delete

command → show databases; → use cats; → select database();

→ show tables;

not correct



~ Show tables

output ↗

Tables-in-cats

cat
cats2
cats3
cats4
people
unique-cats

2/3

-- Create the new cats table:

```
CREATE TABLE cats( cat-id INT AUTO_INCREMENT,  
name VARCHAR(100), breed VARCHAR(100), age INT,  
primary KEY (cat-id)
```

);

-- insert

cat-id	name	breed	age
1	Zingo	tabby	5
2	Cindy	maine coon	11
3	Bumbledore	maine coon	11
4	Egg	persion	4
5	Nasty	tabby	13
6	George Michael	Rangdoll	9
7	Jackson	sphynx	7

Read → How do we retrieve and search data?

SELECT

→ Select name from cats;

→ Select age from cats;

or Select name, age from cats;

... name, breed ...

} For a particular column
→

The where clause

Select * from cats where age = 4;

or
Select name, age from cats where age = 4;

or
Select * from cats where name = 'Egg';

→ case insensitive
Egg
eGg
EGG
etc.

→ Rapid fire questions

- write a sql that selects ① name, breed ② cat_id ③ name, age only for a particular breed ④ cat_id = age

cat_id	name	breed	age
1	ringo	taby	5
2	eindy	main coon	11
3	Dumble dore	main coon	11
4	Egg	pusian	4
5	misty	taby	13
6	George Michel	Rangdoll	9
7	Jackson	Sphynx	7

- Solution →
- (I) Select name, breed from cats
 - (II) Select cat-id from cats
 - (III) Select name, age from cats where breed = 'tabby';
 - (IV) Select cat-id, age from cats where cat-id = age;
✓ output

cat-id	age
4	4
7	7

(Aliases Rename)

Select cat-id as id, name from cats;



→ code ←

use 'AS' to alias a column in your results. (it doesn't actually change the name of the column in the table.)

Select cat-id AS id, name FROM cats;

→ UPDATE ←

Syntax (How do we edit existing data?)

UPDATE cats SET breed = 'shorthair'

WHERE breed = 'Tabby';

* For all things to change remove the last part

→ UPDATE employees SET last-name = 'Toosertbind' ;

id	last-name	first-name	middle-name	age	current-status
1	Toosertbind	Thomas	NULL	87	employed
2	Toosertbind	Thomas	NULL	87	employed
3	Toosertbind	Thomas	NULL	87	employed
4	Toosertbind	Thomas	NULL	87	employed

UPDATE | employees | SET | current-status = 'laid-off' ,
last-name = 'Who cares' ;

id	last-name	first-name	middle-name	age	current-status
1	who cares	Thomas	Null	87	laid-off
2	who cares	Thomas	Null	87	laid-off
3	who cares	Thomas	Null	87	laid-off
4	who cares	Thomas	Null	87	laid-off

Command → UPDATE cats SET age=14 WHERE name = 'Misty' ;

→ Quick Rule of Thumb ←

- if we change Jackson → Jack , change Ringo's breed to 'British shorthair' update both Main coono age to be 12

→ UPDATE cats SET name = 'Jack' WHERE name = 'Jackson';
→ UPDATE cats SET breed = 'Bottish shorthair' WHERE breed = 'tabby';
→ UPDATE cats SET age = 12 WHERE breed = 'maine coon';

→ Delete ←

Syntax → ① Delete FROM cats WHERE name = 'Egg';

OR ② Delete FROM employees;

→ to delete everything

③ Desc employees;

table will be there but the values will be deleted

Q. Delete all 4 year old cats

Delete FROM cats WHERE age = 4;

Q. Delete cats where cat-id = age

DELETE FROM cats WHERE cat-id = age;

Q. Delete all cats

Delete FROM cats;

Section-6

→ CRUD-CHALLANGE ←

- ① Create a database shirts-db
 - ② Create a new table shirts
 - ③ Shirt-id, Article, color, shirt-size, last-worn
↑
• primary key
 - ④ Add a new shirt Purple, size M, last worn 50 days ago
 - ⑤ Select all shirts but print only article, color
 - ⑥ Select all medium shirt but print out everything except shirt-id
 - ⑦ Update all polo shirt to size L
 - ⑧ Update shirt last worn 50 days ago to 300
 - ⑨ Update all white shirts change size to ~~xxs~~ and color to off-white
 - ⑩ Delete all bold shirts, last worn 200 days ago
 - ⑪ Delete all tank tops
 - ⑫ Delete all shirts
 - ⑬ Drop the entire shirts table
-
- ① create database shirts-db;
use shirts-db
select database();
 - ② create table shirts(shirt_id INT PRIMARY auto-increment PRIMARY KEY , article VARCHAR(50) , color VARCHAR(100) , shirt-size VARCHAR(10) , last-worn INT);

insert into shirts (article, color, shirt-size, last-worn)....

- ④ insert into shirts (article, color, shirt-size, last-worn)
values ('Polo shirt', 'purple', 'M', 50);
- ⑤ select * from shirts article, color from shirts;
- ⑥ select * from shirts where shirt-size = 'M';
select article, color, shirt-size, last-worn from shirts
where shirt-size = 'M';
- ⑦ UPDATE shirts SET size = 'L' where article = "polo shirt";
- ⑧ UPDATE shirts SET last-worn = 0 where last-worn = 50;
- ⑨ UPDATE shirts SET shirt-size = "xs", color = "off white"
WHERE color = "white";
- ⑩ DELETE FROM shirts where last-worn = 200;
- ⑪ Delete FROM shirts WHERE article = "tank top";
- ⑫ Delete from SHIRTS;
- ⑬ DROP TABLE shirts;
show tables;
desc ~~table~~ shirts;
show tables;
- empty set

Section-7

→ String Functions ←

→ CONCAT ←

combine data for cleaner output

Select 1+4;

Select 1+4;

5

Select CONCAT ('h', 'e', 'l');

Select 'he' ;

CONCAT ('h','e','l')

he

he

he

✓ books table name

book-id	title	author-fname	author-lname	released-year	stock-quantity
1.	The Namesake	Jhumpa	Lahiri	2003	32
2.	American Gods	Neil	George	2016	43
3.	Just Kids	Don	Eddas	2019	45
4.	Cannery Row	Jhan	Clarks	2011	60
5.	White Noise	David	Stelle	2001	600
..					

Commands → Select **CONCAT**(author-fname, author-lname)
from books;

output

CONCAT(author-fname, author-lname)

JhumpaLahiri

NeilGeorge

DonEddas

..

..

Command → Select CONCAT (author_fname, ' ', author_lname)
as full-name from Books;

Full-name

Jhumpa Lahiri

Neil George

Don Dees

..

command → Select ~~CONCAT_WS (' - ' ,~~ author_fname ,

to include a particular character in all rows

author_lname , title) as bull-dog from Books;

bull-dog

Jhumpa Lahiri - The Namesake

Neil George - American Gods

..

SUBSTRING

works with parts of strings

Select SUBSTRING ('Hello World' , 1,4) ;

SUBSTRING ('Hello World' , 1,4)

Hello

↑ start of string

SELECT SUBSTRING ('Hello World' , 1,6) ;

(1)(6) ;

6 space have been

entry print out

SUBSTRING ('Hello World' , 1,7)

Hello_w
↑ space

(6)(2) → 2 space

→ wo

① SELECT SUBSTRING ('Hello world', 7)
Output → world

② SELECT SUBSTRING ('Hello world', -3)
Output → rld

③ SUBSTR OR SUBSTRING
BOTH WORKS

④ select SUBSTR (title, 1,10) as 'short title' FROM Books;

short title

The Namusa (only 10 characters are printed)

Norse Myth

American G

Inteprete

...

⑤ SELECT SUBSTR(author_fname, 1,1) , author_fname FROM books;

SUBSTR(author_fname, 1,1)

J
N
D
B
J

author_fname

Jhumpa
Nöl
Dave
Button
John

→ combining string functions <

Command → SELECT CONCAT (SUBSTR(title , 1, 10) , '...') AS short-title FROM books;

Command → SELECT CONCAT (SUBSTR(author-fname , 1, 1) , ',' ,
② SUBSTR(author-lname , 1, 1) , ',' ,
) AS author-initials
from Books;

short title

The Name of a ...
Norse Myth ...
American Ch ...
A Hologram ...

??

author-initials

J-L.
N-G.
J-L.
D-E.

REPLACE

Replace parts of strings

↳ string

SELECT REPLACE ('Hello world', 'Hello', 'Hi');

Output → 'Hi world'

replace('Hello world', 'Hello', 'Hi');

Kiwi world

* Actually it didn't update anything any values in the table

→

value

want to

be replaced

↓
Replacing
value

SELECT REPLACE ('chase bread coffee milk', ' ', 'and');

Output → chase and bread and coffee and milk

Commands → SELECT Replace (title, ' ', '-') FROM books;

replace(title, ' ', '-')

The-Nameake

Norse-Mythology

American-Books

Introduction-of-Maladius

REVERSE

SELECT REVERSE ('Hello world');

Output → dlrow olleH

command → SELECT CONCAT ('woof', REVERSE ('woof'));

woof	foow
------	------

command → SELECT CONCAT (author-fname, REVERSE (author-fname))
from books;

	concat (author-fname, '') , reverse (author-fname))
Jhompaa	apmuhJ
Neil	lien
Dave	evAD
:	:

CHAR LENGTH

counts characters in a string

Command → SELECT CHAR_LENGTH ('Hello-World');

Output → 11

SELECT LENGTH ('somechineseSymbol, ');

LENGTH('些字, ')
6

gt works on bytes occupied

SELECT CHAR_LENGTH ('');

Output → (2) ✓

command → Select CONCAT ('woof', REVERSE ('woof'));

woof foow

command → SELECT CONCAT (author-fname, REVERSE (author-fname))
from books;

concat (author-fname, ' ', reverse(author-fname))

Jhumpa apmuhJ

Neil lieN

Dave evAD

⋮

CHAR LENGTH

counts characters in a string

Command → SELECT CHAR_LENGTH ('Hello World');

output → 11

SELECT LENGTH('somechinese symbol, 𠮾');

LENGTH('𠮾')	6

it works on bytes occupied

SELECT CHAR_LENGTH ('');

Output → 2 ✓

→ SELECT CHAR_LENGTH(title) as length, title From bookg;

length	title
12	The Narniaca
15	Norse Mythology
13	American Gods
23	Interpreter of Maladies
..	

→ SELECT author-name , char-length (author-name) as
‘length’ from books;

Author-Name	length
Lahiri	6
Craimam	6
Craimam	6

→ select CONCAT (author-lname, ' is ', CHAR_LENGTH
• (author-lname), " characters long")
From books;

concat (author_name, \$16^2)

Lahizi is 6 characters long

Gelman is 6 characters long

1

UPPER() AND LOWER()

change a string's CASE

Select UPPER('Hello')

UPPER('Hello')

HELLO

Select LOWER('Hello')

LOWER('Hello')

hello

UPPER/ UCASE

LOWER/ LCASE

Q. Point title I LOVE THE NAMESAKE !!! ?

like this

Select concat ('I LOVE', UCASE(title), '!!!')
from books;

concat ('I LOVE', UCASE(title), '!!!')

I LOVE THE NAMESAKE !!!

I LOVE THE HORSE MYTHOLOGY !!!

...

...

INSERT

insert character & place
करता है।

- ① `INSERT ('Hello Bobby', 6, 0, 'There');`
- (6) ↗
= ↗ position वाले के बीच से 6वां अक्षर
0 ↗ word to be inserted

Output → Hello Therebby

- ② `SELECT INSERT ('Hello Bobby', 6, 4, 'There');`

Output → Hello Thereby

- ③ `SELECT INSERT('Hello Bobby', 6, 6, 'There');`

Output → Hello There

LEFT / RIGHT

3 character left off

- `SELECT LEFT ('omghahahaha!', 3);`

Output → omg

- `SELECT RIGHT ('omghahahaha!', 4);`

Output → lol!

REPEAT

- `SELECT REPEAT ('ha', 4);`

Output → hahahaha

TRIM

- ① SELECT TRIM (' boston ') ;
output → [boston] spaces removed at end but not at middle
- ② SELECT TRIM (' San Antonio ') ;
output → San V Antonio
- ③ SELECT TRIM (LEADING .) FROM '..... san antonio.';
- Output → San Antonio..
- ④ SELECT TRIM (TRAILING .) FROM '.....; SAN Antonio..';
output → san antonio
- ⑤ SELECT TRIM (BOTH .) FROM ' san Antonio..';
output → San Antonio

Exercise

19. Reverse and uppercase the following sentence
'why does the cat look at me with such hatred ?'
20. SELECT REPLACE (CONCAT ('I', ' ', 'Lila', ' ', 'cats'),
' ', '_') ;
30. Replace spaces with in titles of books
0. Point author F-name in forward column with reverse Fname in backward column
0. Point full name in CAPS
0. point title and release year in the form of sentence like
'The movie was released in 2003 ...'

7 Q. Print the book title and the length of each book title

8 Q.

short title	author	quantity
American Cr...	Chadman, Neil	12 in stock
A Heartbreak...	Eggins, Dave	104 in stock

Solution

- ① SELECT REVERSE(UPPER('WHY DOES MY CAT LOOK AT ME WITH SUCH HATRED?'));
- ② SELECT REPLACE(title, ' ', '->') FROM Books;
- ③ SELECT author_lname AS forwards, REVERSE(author_lname) AS backwards FROM books;
- ④ SELECT CONCAT(author_fname, ' ', author_lname) AS 'full name in caps' FROM books;
- ⑤ SELECT CONCAT(title, 'was released in', released_year) AS blurb FROM books;
- ⑥ SELECT title AS title, CHAR_LENGTH(title) AS 'character count' FROM books;
- ⑦ SELECT CONCAT(LEFT(title, 10), '...') AS short_title, CONCAT(author_lname, ', ', author_fname) AS author, CONCAT(stock_quantity, ' ', 'in', ' stock') AS quantity FROM books;

Section 8

| Refining Sections |

DISTINCT

SELECT DISTINCT author_lname FROM books;

Replace all repeated author_lname elements to one element

Command → Select author_fname from books;
→ Select Distinct author_lname from books

author_lname	author_lname
Lahiri	Lahiri
Gaiman	Gaiman
Gaiman	
Lahiri	
Eggens	Eggens
Eggens	

Command → Select distinct ~~with~~ CONCAT(author_fname, ' ', author_lname)
FROM books;

→ Concatenation

Thuma Lahiri	Thuma Lahiri
Nil Gaiman	Nil Gaiman
Nil Gaiman	
Dave Eggens	Dave Eggens
Dave Eggens	

Command → Select distinct author_fname, author_lname, released_year
from books;

①

author_fname	author_lname	released_year
Shivam	Kumar	2003
Shivam	singh	2003

(X)

X all three values need to be same.

②

N

shivam	kumar	2004
Shivam	kumar	2004

This will be filtered 😊

ORDER-BY

Sorting our result

Command →

SELECT author-lname FROM books ORDER BY

author-lname ;

- It ASCENDS By Default $a \rightarrow z$

- But we can change that by DESC ;

Command → ↴

author-lname DESC ;

Numbers too !

SELECT title, pages FROM books ORDER BY pages ;

title	Pages
what we talk about..	176
Cannery Row	181
Inhalation of Maladious	198
..	..
..	..

① select author_lname from books; (by default ASC)

② select author_lname from books order by author_lname;

Lahiri	
Chaiman	
Chaiman	
Eggins	
Eggins	
Chabon	
smith	
Carver	

Command → Select book-id, title, author-Fname, author-Lname,
released-year From books ORDER by released-year;

book_id	title	author_name	author_lname	released_year
14	Cannery Row	John	Steinbeck	1945
11	What We Talk About..	Raymond	Carver	1981
13	White Noise	Don	Dilillo	1985
12	Where I'm Calling From..	Raymond	Carver	1989
4	American Gods	Neil	Gaiman	2001
7	..			
3	..			

Command \rightarrow ① $\underline{\text{author_fname}}$, ② $\underline{\text{author_lname}}$, ③ $\underline{\text{released_year}}$
 \rightarrow Select title, author_fname, author_lname FROM books
 ORDER BY ②;

इस 2 का नियम हैं ~~author~~ from author-fname
 according arrange कर।

Command \rightarrow SELECT author_fname, author_lname FROM books ORDER By author_lname, ~~author_fname~~, ~~released_year~~

author-fname	author-lname	released-year
Raymond	Carver	2003
Raymond	Carver	2016
Don	Dellilo	2019
Dave	Eggers	2018
Null	Eggers	2001

Command \rightarrow SELECT ?? ORDER By author_lname, released-year;

author-fname	author-lname	released-year
Raymond	Carver	2003
Raymond	Carver	2013
Don	Dellilo	2019
Dave	Eggers	2001
Null	Eggers	2018

* Second Priority goes to released year after author-lname

LIMIT

→ controls no. of results we get back

SELECT title, released_year FROM books ORDER BY released_year
DESC LIMIT 5;

..	..
1	
2	
3	
4	
5	

} 5 data is printed

→ Select book-id, title From books limit 0,10;

book-id	title
1	The Nomadic 0
2	Norse Mythology 1
3	American Gods 2
4	Gintyptur of Maladius 3
5	A Hologram For the King 4
6	The Circle 5
7	The Amazing Adventures of 6
8	Dust Kids 7
9	
10	

Starting from

→ Select book-id, title From books limit

book-id	title
4	Gintyptur of Maladius
5	A Hologram For the King

3) 4 → No. of values want to printed

0 means starting

LIKE

→ Better Searching

Q) HERE author-fname LIKE '%, day%'
wild cards

Command → SELECT title, author-fname, author-lname FROM books
where author-fname LIKE '%, day%, %'

title	author-fname	author-lname
A hologram for the king	Dave	Eggus
The circle	Dave	Eggus
A Heartbreaking work of Oblivion : Stories	Dave	Eggus
Consider the Lobster	David	Wallace
Fake-books	David	Wallace
	Don	Harris
	Frida	Harris

→ Search books F-name में 'da' की भी ही संकेत है सबको display करेगा।

% sign का use तरफ जैसे जाना detail की मायग्रा हो
accurate detail अंगारे पर्हा है ही — (undiscrose)
use करो  !

Command → Select * from books where title like '% : %';
→ output → Empty set

Command → Select * from books where title like '% : %';

book-id	title	author-fname	author-lname	... Pages
5	A hologram... $\text{!} \text{ A Novel}$	Dave	Eggers	352
11	What we... $\text{!} \text{ love stories}$	Raymond	Carver	176
12	"	"	Carver	526
13	Oblivion $\text{!} \text{ Stories}$	David	Foster	329

असी कॉर्स $\leftarrow [\% : \%]$ \rightarrow किंतु कॉर्स character
character

commands select * from books where title like '%ected';

book-id	title	author-fname
12	.. from: Selected stories	"	"	"	"

command \rightarrow select * from books where title author-fname
like '-----';

↳ 4 characters का कॉर्स
author-fname #;

book-id	title	author-fname
"	"	<u>Nil</u>	"	"	"
"	"	Nil	"	"	"
"	"	Dave	"	"	"
"	"	Dave	"	"	"
"	"	<u>Nil</u>	"	"	"
"	"	<u>John</u>	"	"	"

command \rightarrow Select * from books where author-fname like
' - a - ';

book-id	title	author-fname
"	"	<u>Dan</u>	"	"

Select * from books where author-fname like %y.%

Meaning last of ~~02~~
n starts

book-id	...	author-fname
13	..	Don	..	49
14	..	Jhon	..	95
17	..	Dan	..	29

- Q. But what if gom searching for a books with a % in it
 ↳ book with an _ in it

Solution when title like %_%.% or %_.%

%_.%

→ Exercise ↵

10. Select all stories collections , titles that contain 'stories'
20. Find the longest books point out the title and page count
30. print summary for the title and year for the three most recent books
40. find all the books that contains author-fname with space (' ')
50. Find the 3 books that have lowest Stock quantity
 Select, title year stock-quantity
60. Print title and author-fname , sorted first by author-fname and then by title
70. For every single author sorting by last name

Q.1

title

what we talk about when we talk about Love: stories
when I'm calling From: selected stories

Oblivion: stories

Ans → Select title from books where title like '%. Stories%';

Q.2

title

pages

The Amazing Adventures of Kavalin & clay.

634

ans → select title, pages from books ORDER BY pages desc limit 1;

Q.3

Summary

Lincoln in the Bardo - 2017

Norse Mythology - 2016

10%. Happiness - 2014

Ans → Select concat (title, ' - ', released - year) as summary
from books order by released - year desc limit 3 ;

Q.4

title	author_lname
Oblivion: stories	Foster Wallace
Consider the Lobster	Foster Wallace

Ans → select title, author_lname from books where author_lname
like '%. %. %';

Q.5

title	released_year	stock_quantity
American Gods	2001	12
when I'm calling From..	1989	12
what we talk about..	1981	23

Ans → select title, released_year, stock_quantity from books
order by stock_quantity limit 3 ;

Q.6

title

what we talk about when we talk

where I'm calling from:

The amazing . . .

white Noise . . .

author-lname

Carver

Carver

Chabon

D.Lillo

Ans) Select title, author-lname from books order by author-lname, title;

Q.7

yell

My FAVORITE AUTHOR is RAYMOND CARVER!

My FAVORITE AUTHOR is MICHAEL CHABON

Select concat ('My FAVORITE Author is ', author-fname, ' ',
author-lname) as yell from books
order by author-lname;

Section-9

Aggregate Functions

COUNT

Command → SELECT COUNT (*) FROM BOOKS;

→ Counts No. of Rows

COUNT(*)
19

command → select title, author-fname, COUNT(*) FROM books;

X

Q. How many author-fnames?

Ans → select count (author-fname) FROM books;

• It counts everytime a value is present in author-fname value column

① Grant into books () values (); X 2

② ~~Count~~ select count (*) FROM books;
Output → 21

③ select count (author-fname) FROM books;
Output → 19

Q. How many distinct author-fnames?

Ans → select COUNT(DISTINCT author-fname) FROM Books;
Output → 12

Q. How many titles contain 'the'?

Ans → select title FROM books where title like '%.the%';
↳ दर्शन 6 दूर 6

✓ " Count(*) → Output → 6

GROUP-BY

" GROUP By summarizes or aggregates identical data into single rows "

command → SELECT author_lname FROM books GROUP BY author_lname

* behind the scenes

title	author-lname
The Namesake	Lahiri
Norse Mythology	Haiman
American Gods	Craigman
Interpreter of Maladies	Lahiri
A Hologram for the King : A Novel	Eggars

Groups Formed ↴

②	The Namesake	Lahiri
②	Interpreter of Maladies	Lahiri

②	Norse Mythology	Haiman
②	American Gods	Craigman

①	A Hologram For the King : A Novel	Eggars
---	-----------------------------------	--------

Q. How many books each author has written

SELECT author_lname, count(*) FROM books
GROUP BY author_lname;

command → select * author-name , count(*) as books-written
FROM books GROUP BY author-name ORDER by books-written DESC;

author-name	books-written
Ozaiman	3
Eggers	3
Lahiri	2
..	..

Example-2 ① Select released-year from books group by released year;

released-year
2003
2016
2001
1996
2012
..

② Select released year , count(*) From books GROUP BY released-year;

released-year	Count(*)
2003	2
2016	1
2001	3
1996	1
..	..

~~(X)~~ SELECT author-name , title From books GROUP BY author-name;

count display the two titles simultaneously for each element

MIN AND MAX

Q. what is the minimum released year from books?

ans \Rightarrow SELECT MIN(released-year) FROM books;

MIN(released-year)
1945

Q. find maximum no. of pages?

ans \Rightarrow SELECT MAX(pages) FROM books;

Max(pages)
634

Q. what about for alphabatus?

ans \Rightarrow SELECT MIN(author-lname), MAX(author-lname) FROM books;

* work according to lowest and highest alphabatus

Min(author-lname)	Max(author-lname)
Carver	Steinbeck

Q. What if i want the title of the longest book?

X Select MAX(pages), title from books;
not working why? 😐

(Subquery)

②
 SELECT * FROM books
 WHERE pages = (SELECT MIN(pages))
 OR pages = (SELECT MAX(pages))
 OR pages = (SELECT COUNT(*) FROM books);

this query will be stored and evaluated then

book-id	Title	" "	" "	" "	pages
7	The Amazing Adventures				634

check → insert into books (title, ^{page}books) values ('Raja Billa Shivam', 634);

Q. What if the row has more than one data of page 637 ?

Command → Select title, pages from books order by pages desc limit 1;

title	pages
Raja Billa Shivam	634

Not work properly
misses data

Command → SELECT * from books

WHERE pages = (SELECT Max(pages) FROM books);

✓ → displays all two columns:

→ GROUPING BY MULTIPLE COLUMNS

Command → Select author-lname, count(*) from books group by author-lname;

Author-lname	count(*)
Lahiri	2
Nauman	3
Leggett	3
Smith	1
Conver	2
D'Lillo	1
Steinbeck	1
Foster Wallace	2

Select author-fname, author-lname , count(*) From books

Group by author-lname , author-fname;

~~star attr value same for group if~~
~~group |~~

author-fname	author-lname	count(*)	Crouping as multi value
Jhumpa	Lahiri	2	
Nil	Gaiman	3	
Dave	Eggers	3	
Patti	Smith	1	
Raymond	Carver	2	
Don	DeLillo	1	
Don	Harris	1	
Frida	Harris	1	
Null	Null		

Command → SELECT CONCAT(author-fname, ' ', author-lname)
as Author , count(*) From books Group by author;

Crouping
as one value

Author	Count
Jhumpa Lahiri	2
Nil Gaiman	3
Dave Eggers	3
Don Harris	1
Frida Harris	1

| MIN/ MAX WITH GROUP BY |

Q. find the year each author published their first book ?

command? Select author_lname

```

    COUNT(*) AS books_written,
    MAX(released_year) AS latest_release,
    MIN(released_year) AS earliest_release,
    MAX(pages) AS long_page_count,
FROM Books GROUP BY author_lname;
  
```

प्रेरणा कीटा
उस sub group
के लाई तो
details की

Q. Select author_lname, MIN(released_year) FROM books GROUP BY author_lname;

author_lname को group करके condense करके एक जाएंगा और
जिस से multiple value की ओर से शर्करा की जाए (Min) value होगा।

Q. Select author_lname, MAX(released_year), MIN(released_year)
FROM books group by author_lname;

SUM

{ add things together }

Q. SELECT SUM(pages) FROM books;

SUM(pages)
7257

Select sum/author_lname
FROM books;

Output → 0

Q. Select author_lname, SUM(pages) FROM books GROUP BY author_lname;

author_lname	SUM(pages)
Lahiri	489
Haiman	977
Eggers	1293
..	..

Lahiri has written a total
of 489 pages
(works for subgroups in this
case)

- ① SELECT SUM(pages) FROM Books;
- ② SELECT author_lname , count(*) , SUM(pages) FROM books
GROUP BY author_lname;

AVG

Q. calculate the average released year across all books?

- ① SELECT AVG(released-year) FROM Books;

or
Output → 1999.7895

- ② SELECT AVG(pages) FROM Books;

Output → 362.85

Q. calculate the average stock quantity for books released in the same year?

SELECT released-year , AVG(stock-quantity) FROM books GROUP BY released-year;

released-year	Avg (stock-quantity)	COUNT(*)
2003	66	2
2016	43	1
2001	134.33	3
1996	97	1

Total no of rows and values
of 4 ref to

→ Exercise ←

Q.1. Point the number of books in database ?

→ select count(*) From books

Q.2 Point how many books released each year ?

→ select released-year, count(*) FROM books group by released-year

Q.3 point out total no. of books in stock

→ select sum(stock-quantity) From books;

Q.4 Find out the average released year for each author ?

select concat(author-fname, ' ', author-lname)
, avg(released-year) from books group by author-
fname, author-lname;

concat(author.....)	avg(released-year)
Jhumpa Lahiri	1999.5000
Muhammad Ali	2006.6667
..	
Michael Chabon	2000.000

Q.5 Find the full name of the author who wrote the longest book

select concat(author-fname, ' ', author-lname) From books where pages = (max select max (Pages) From books);

Q.6 →

Make this happen

Year	# books	avg pages
1945	1	181.000
1981	1	176.000
..
2001	3	249.5000
..
2017	1	367.000

select released_year as year, count(*) as '# books'
, avg(pages) as 'avg pages' from books group
by released_year order by released-year ;

Section-10

① VARCHAR VS CHAR

CHAR \rightarrow it has a fixed length

command
 CREATE TABLE status (abbr CHAR(2));
 insert INTO status (abbr) values ('CA'), ('NY');
 select * from status;

abbr
CA
NY

optimized to store 2 characters
 by taking up least memory
 only to store two characters

2 rows in set

say

if i write only 1 character "x", then padding will be done like "x_" using whitespace character and then it is stored when we call back the process will be reversed, padding will be removed and we get the same value "x".

CHAR(3) \rightarrow only 3 characters are allowed

if < 3 characters, whitespace character will be carried out.

Now a table

Value	Char(4)	Storage	Varchar(4)	Storage
'r'	'r'	4 bytes	'r'	1 byte
'ab'	'ab_'	4 bytes	'ab'	3 bytes
'abcd'	'abcd'	4 bytes	'abcd'	5 bytes

§. Some examples where CHAR is used.

State Abbreviations: CA, NY

Y/N/NO Flags Y/N

ZIP codes 59715, 94924

→ Otherwise USE VARCHAR 

NUMBERS			
	Storage	minimum signed value	maximum unsigned value
TINYINT	1	-128	127
SMALLINT	2	-32768	32767
MEDIUMINT	3	-8388608	8388607
INT	4	-2147483648	2147483647
BIGINT	8	- 2^{63}	$2^{63} - 1$

minimum unsigned value	maximum value unsigned
0	255
0	65535
0	16777215
0	$2^{64} - 1$

Command Create table Family (children TINYINT UNSIGNED)
Insert into Family (children) values (-3)

Output

→ Error Minimum value entering should be 0.
→ Out of range value

DECIMAL

Insert into Family (children) values (1.5), (90.008)

Select * from Family

Output

2 } Rounded to whole no.
90 =

Q. How to store a decimal no. ? total no. of digits
syntax → DECIMAL ($\overset{5}{\text{5}}$, $\overset{2}{\text{2}}$)
 ↑
 ↓
 digits after decimal

Example:

→ 5 digits

→ 999.99

→ 2 digits

this is the maximum no. that can be stored of this type: DECIMAL (5,2)

CREATE TABLE products (price) VALUES DECIMAL (5,2);

INSERT INTO products (price) VALUES (4.50);

output → price 4.50

INSERT INTO products (price) VALUES (456.99)

output → 456.99

INSERT INTO products (price) VALUES (8347.1);

output → OUT OF RANGE ERROR

INSERT INTO products (price) VALUES (5.026);

output → 1 row affected, 1 warning ; output → 5.026

SHOW WARNINGS;

data truncated for column 'price' at row 1

Excluding maximum decimal value will truncate data but exceeding the significant digit will give an error

FLOAT AND DOUBLE

* It stores large

- Q. Why should we use float/Double instead of decimal ?
 Ans. To store large numbers float/~~double~~ double are used instead of decimal but it comes with a cost of precision (accuracy).

Data type	Memory used	Precision
FLOAT	4 bytes	~7 digits
DOUBLE	8 bytes	~15 digits

- ① create table nums (x FLOAT, y DOUBLE);
- ② insert into nums (x,y) values (1.2345, 1.2345)

x	y
1.2345	1.2345

③

→ Values (1.12345678, 1.12345678)

x	y
1.12346	1.12345678

Precision
Started
after 7 digits

* For higher precision use DECIMAL , FOR ...

DATES & TIMES

DATE → • values with a date but no time
• 'yyyy-MM-DD' Format

TIME → • values with a time but no date
• 'HH:MM:SS'

DATE TIME → • values with a date and time

• 'yyyy-MM-DD HH:MM:SS' Format

→ working on DATE, TIME, DATETIME

```
CREATE TABLE people( name VARCHAR(100)  
                    birthdate DATE,  
                    birthtime TIME,  
                    birthdt DATETIME  
                    );
```

```
INSERT INTO people( name, birthdate... )
```

```
VALUES('Elton', '2000-07-02', '11:00:00', 2000-07-02 11:00:00);
```

↑
Space

```
SELECT CURTIME(); / CURRENT_TIME
```

Output → 14:55:40

```
SELECT CURDATE(); / CURRENT_DATE
```

Output → 2023-09-27

```
SELECT NOW(); / CURRENT_TIMESTAMP()
```

Output → 2022-10-05 14:56:06

Command \Rightarrow INSERT INTO people1(name, birthdate, birthtime, birthdt)
 birthdt) value ('^{Hajd}'CURDATE(), CURTIME(), NOW());

name	birthdate	birthtime	birthdt
Lulu	1985-04-11	09:45:10	1985-04-11 09:45:10
Elton	2000-12-25	11:00:00	2000-12-25 11:00:00
Juan	2020-08-15	23:59:00	2020-08-15 23:59:00
Hajd	2023-09-27	11:14:22	2023-09-27 11:14:22

→ Some Useful Date Functions

DAY() Day of Month

→ Select birthdate, DAY(birthdate) FROM people;

birthdate	DAY(birthdate)
2000-12-25	25
1985-04-11	11

→ Select birthdate, DAY(birthdate),
 DAYOFWEEK(birthdate) FROM people

		Day of week(..)
..	..	2
		5
		7

DAYOFWEEK

1 = Sunday

...

7 = Saturday

> Monday

> Saturday

- DAYOFYEAR (range 1 → 366)
- MONTHNAME → name of Month

SELECT name, birthdt, YEAR(birthdt), MONTHNAME(birthdt)
FROM people;

name	birthdt	YEAR(birthdt)	MONTHNAME(birthdt)
Elton	2000-12-25 11:00:00	2000	December
LULU	1985-04-11 9:45:00	1985	April
Juan	2020-8-15 23:59:00	2020	August
Hazl	2022-10-05 14:58:00	2022	October

* If you are collecting information from birthtime, which don't have month, date value it will by default extract from current date, year value.

TIME FUNCTIONS

- HOUR, MINUTE, SECOND

Select hour(birthtime), HOUR(birthtime) From people;

birthtime	HOUR(birthtime)
11:00:00	11
9:45:00	9
23:59:00	23
14:58:00	14

FORMATTING DATES

date is 1985-04-11 → April 11 1985

Q. How to format like that?

option 1 → Select concat(Monthname(birthdate), ' ',
DAY(birthdate), ' ', YEAR(birthdate)) from people;

MONTHNAME(birthdate)	DAY(birthdate)	YEAR(birthdate)
December	25	2000
April	11	1985
August	15	2020
October	5	2022

Output → December 25 2000

command → SELECT birthdate, DATE_FORMAT(birthdate,
'%a %b %Y') FROM people;

DATE FORMAT

birthdate	DATE_FORMAT(birthdate, '%a %b %Y')
2000-12-25	Mon Dec 25th
1985-04-11	Thu APR 11th
2020-08-15	Sat Aug 15th

SELECT birthdt ; DATE_FORMAT (birthdt , '%Y') FROM people;

Output →

2000-12-25 11:00:00 → 11:00:00 AM

.. 23:59:00 .. 11:59:00 PM

command → SELECT birthdt , DATE_FORMAT (birthdt ,
⑥ BORN ON %Y ⑦) FROM people;

birthdt	DATE FORMAT (birthdt , 'BORN ON: %Y')
2000-12-25 11:00:00	BORN ON: 11:00:00 AM
1985-04-11 09:45:10	BORN ON: 09:45:10 AM
..	..

DATE MATH

DATEDIFF (Expr1 , Expr2)

↳ takes date/ date-time and do operations

birthdate

Command → SELECT , DATEDIFF(CURDATE() , birthdate)
FROM people ;

birthdate	DATEDIFF(CURDATE() , birthdate)
2000-12-25	7954
1985-04-11	13691
2020-08-15	781
2022-10-05	0

DATE_ADD / DATE_SUB

SELECT DATE_ADD ('2018-05-01', INTERVAL 1 DAY);

Output → 2018-05-02

SELECT DATE_SUB ('2018-05-01', INTERVAL 1 YEAR);

Output → 2017-05-01

SELECT DATE_ADD('2020-12-31', 23:59:59);

INTERVAL 1 SECOND);

command → SELECT birthdate, DATE_ADD(birthdate,
INTERVAL 18 year) ~~getdate~~

birthdate	DATE-ADD(birthdate, ^)
2000-12-25	2018-12-25
1985-04-11	2003-04-11
..	..
2020-08-15	2038-08-15

SELECT CURTIME();

TIME DIFF

CURTIME()

17:31:55

SELECT TIMEDIFF(CURTIME(), '07:00:00');

TIMEDIFF(CURRENTTIME(), '07:00:00')

10:32:13

Ex. ~~SELECT NOW() - INTERVAL 18 YEARS;~~

Now() - INTERVAL 18 YEARS
2004-10-05 17:36:10

ज़मी से 18 वर्ष
पहुंचा।

Ex. SELECT name, birthdate, birthdate + INTERVAL 21
YEARS FROM people;

name	birthdate	birthdate + INTERVAL 21 YERS
Elton	2000-12-25	2021-12-25
Lulu	1985-04-11	2006-04-11
Juan	2020-08-15	2041-08-15

Ex. SELECT name, birthdate, YEAR(birthdate + INTERVAL
21 years) as will_be_21 from people;

name	birthdate	will_be_21
Elton	2000-12-25	2021
Lulu	1985-04-11	2006
Juan	2020-08-15	2041
Hazel	2022-10-05	2043

TIMESTAMPS

- Contains both date and time parts
- It takes less storage as compared to datetime
- It has a range of 1970-01-01 → 2088-01-19
whereas DATETIME 1000-01-01 → 9999-12-31

Command → CREATE TABLE captions (text VARCHAR (150),
created_at TIMESTAMP default CURRENT_TIMESTAMP
);

Commands → INSERT INTO captions (text) VALUES ('Just me
and the girls chillin');

INSERT INTO captions (text) VALUES ('beautiful
sunset');

Select * From captions;

text	created_at
Just me and beautiful girls chillin	2022-10-05 18:04:01
beautiful sunset	2022-10-05 18:04:09

⑥ CREATE TABLE captions2 (text VARCHAR(150),
created_at TIMESTAMP default CURRENT_TIMESTAMP,
updated_at TIME STAMP ON UPDATE CURRENT_TIMESTAMP);
→ insert into captions(text) value ('beautiful sunset');

text	Created-at	Updated-at
beautiful sunset	2023-09-27 21:56:07	NULL

→ UPDATE captions2 set text = 'famu' where text
= 'beautiful sunset';

text	Created-at	Updated-at
famu	2023-09-27 21:56:07	2023-09-27 21:58:52

(Exercise)

Q.1 ✓ what's good use case for char?

Q.2 ✓ Fill in the blanks

CREATE TABLE inventory (

item_name _____,

price _____,

quantity _____);

Q.3 what's the difference b/w date time and timestamp;

Q.4 Point out the current time.

Q.5 Point out the current date (but not time).

Q.6 Point out the current day of the week as a number

Q.7 Point out the current day name of the week.

Q.8 Point out the current day and time using this format
mmj dd/ yyyy

Q.9 Point out the current day and time using this format
January 2nd at 3:15
April 1st at 10:18

Q.10 Create a tweet table that stores :

- The tweet content (180 characters)
- A username
- Time it was created

④ Select curtime();

⑤ Select curdate();

⑥ Select now(), DAYOFWEEK(NOW());

⑦ Select NOW(), DATE_FORMAT(NOW(), '%.D');

⑧ Select CURDATE(), DATE_FORMAT(CURDATE(), '%m-%d-%Y');

⑨ Select now(), DATE_FORMAT(NOW(), '%M %D at %H:%i');

⑩ Create table tweet (content varchar(100), username varchar(100), created_at TIMESTAMP default CURRENT_TIMESTAMP);

Grant into tweets (content, username) values ('this is my first tweet', 'colt scat');

content	username	created_at
this is my first tweet	colt scat	2023-09-27 22:53:26

Section-11

LOGICAL OPERATORS

Q. How to ask and get result like this

- ① Select all books not published in 2017
- ② Select all birthdays between 1990 and 1992 ??
- ③ Select all items that are in stock AND price below \$ 19.99

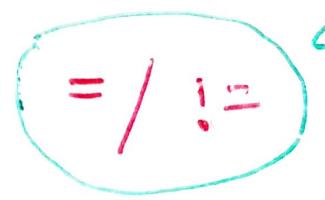
 \Rightarrow Not Equal

command \Rightarrow select title, author_lname FROM books WHERE author_lname = 'Gaiman';

title	author_lname
Norse Mythology	Gaiman
American Gods	Gaiman
Coriolan	Gaiman

); command \Rightarrow .. author_lname != 'Gaiman';

title	author_lname
All books are there except 'Gaiman' books	

 $= / !=$

both are logical operators

NOT LIKE

command \rightarrow SELECT title FROM books WHERE title NOT LIKE '%, %';
Space

title
Norse Mythology
American Gods
::

% points all title containing a space.

command \rightarrow SELECT title FROM books WHERE title

NOT LIKE '%, %';

Space

title
coraline
Fair Book

Example-2 \rightarrow Select * from books where title not like '%, %';

GREATER THAN

Q. How to look for all books which were released after 2000?

command \rightarrow Select * from books where released_year > 2000;

Ex-2 \rightarrow Select * from books where pages > 500;

Q. what will this evaluate

Select 80 > 40; Output \rightarrow 1

Select 100 > 120; Output \rightarrow 0

* It works on that logic to evaluate and point

LESS THAN

command → select title, released-year from books where released-year < 2000;

less than
or equal →

\leq

\geq

→ greater than or
equal to

to

$<=$, \geq , $<$, $>$, $=$, $!=$ all are comparison operators

LOGICAL AND

' $\&$ '

g. Select books written by Dave Eggers, published after the year 2010?

Command →

```
| Select * from books
| WHERE author-lname = 'Eggers'
|   AND
| Select * from books
| WHERE released-year > 2010;
|
```

↓ desired output

Select * FROM books

WHERE author-lname = 'Eggers'

AND released-year > 2010;

Output →

title	author-lname	released-year
A Hologram for the King : A Novel	Eggers	2012
The circle	Eggers	2013

Example 2 → Select title, author-fname, released-year
 From books WHERE released-year > 2010
 AND author-lname = "Eggers"
 AND title like '%. novel %';

Output → Analog Tales for . . . Eggers 2012

Q. How to select title where length/word count of title > 30?
 ans → Select title From books where char-length(title)
 > 30;

pages > 500 ?

ans → ?? > 30 and pages > 500;

any one needs to be true
to both to be true **LOGICAL OR** 0-1 or ✓

Select title, author-lname, released-year FROM books
 where author-lname = "Eggers" **OR** released-year > 2010

title	author-lname	author-fname
Norse Mythology	Gaiman	2016
..	Eggers	2012
..	Eggers	2013
..	Eggers	2001
..	Harris	2014
	Saunders	2017

BETWEEN

Q. How we can select all books published between 2004 and 2015 ?

SELECT title, released-year FROM books WHERE released-year \geq 2004 AND released-year \leq 2015 ; *(2004 and 2015 included)*

title	released-year
A Hologram for a King..	2012
The Circle	2013
Just Kids	2010
Oblivion : stories	2004
Consider the Lobster	2005
10% Happier	2014

Between X and Y (inclusive)

SELECT title, released-year FROM books

WHERE released-year BETWEEN 2004 AND 2015 ;



works the same



NOT BETWEEN X AND Y

Select title, released-year FROM books

WHERE released-year NOT BETWEEN 2004 AND
SPAN 2015 ;

COMPARING DATES

Q. How to get name of people who were born before 2005?

query \Rightarrow select * From people WHERE birthdate < 2005-01-01
but this is a string how sql can compare this (?) ?

Output

because SQL is SQL, hence

name	birthdate	birthtime	birthdt
Elton	2000-12-25	.	??
Lulu	1985-04-11	.	??

(guruuud)

Method-2 \Rightarrow Select * From people WHERE YEAR(birthdate) < 2005;

Q. How to get the people who were born after 12 in the morning?

ans \Rightarrow Select * From people WHERE birthtime > '12:00:00'

name	birthdate	birthtime	birthdt
Juan	..	23:59:00	..
Hazel	..	14:58:29	..

Method-2 \checkmark Select * From people WHERE HOUR(birthtime) > 12;

Some output

CAST

{ More on CHAT-GPT y
Find it

converts one type to other type

Syntax → Select * FROM people WHERE birthtime
BETWEEN CAST('12:00:00' AS TIME)

AND CAST('16:00:00' AS TIME);

this is the technically correct method

Method 2 → Select * FROM people WHERE HOUR
(birthtime) BETWEEN 12 AND 16;
it is short and simple and best Method

THE IN OPERATOR

Q. How to get if we want like this, Select all books written
by caron or Lahiri or Smith?

command → Select title, author_name FROM books
WHERE author_name = 'caron'
or author_name = 'Lahiri'
or author_name = 'smith';
or ...

THE IN MAKES MUCH SHORTER

Select title, author_name FROM books

where author_name IN ('caron', 'Lahiri', 'smith');

Output will be same



Title	Author - Name
The Namesake	Lahiri
Interpreter of Maladies	Lahiri
Just Kids	Smith
What We Talk About	Carver

SIMILARLY **NOT IN** WORKS

- Q. give really suspicious and have a problem with even numbers. Haha, now select all books not published in 2000, 2002, 2004, ..., 2016. ?

Select title, released year FROM books WHERE

released-year != 2000 AND

released-year != 2002 AND

...

Holy shit 😭 ;

→ There is a better way

Select title, released year - from books

where released-year > 2000

AND released-year NOT IN

(2000, 2002, 2004, ..., 2016) ;

→ Wait, there is still a better way

MODULO { Remainder Operator }

Select 10%4 ;

Output → 2

Select 17%6

Output → 5

SELECT title, released-year FROM books
 WHERE released-year >= 2000
 AND released-year % 2 != 0 ;

output →

title	released-year
The Namusake	2003
American Gods	2001
A Heartbreaking ...	2013
coraline	2001
..	..

CASE

→ CASE STATEMENTS ↵

SELECT title, released-year,

CASE

when released-year >= 2000 Then 'Modern Lit'

EELSE '20th century lit'

END AS GENRE

FROM Books;

title	released-year	genre
The Namusake	2003	modern lit
Norse Mythology	2016	modern lit
American Gods	2001	modern lit
Interpreter of Maladies	1996	20th century lit
A Hologram For the King : A NOVEL	2012	modern lit
.....	2013	modern lit
The circle	2000	modern lit
..

Example 2) Select title, stock_quantity,

① ②
No commas
=

CASE

when stock_quantity <= 40 Then *
when stock_quantity <= 70 Then **
when stock_quantity <= 100 Then ***
when stock_quantity <= 140 then ****
ELSE *****

END AS stock

FROM

books;

title	stock_quantity	stock
The Nameless	32	*
Norse Mythology	43	**
American gods	12	*
...	97	***
...	154	****
...	26	*
		..

(NULL)

working with null values

Select * from books where author_lname

IS NULL;

Author-lname Stock-quantity

book_id	title	author_fname	author_lname	author_lname	stock_quantity	pages
20	NULL	NULL	NULL	NULL	NULL	NULL
21	NULL	NULL	NULL	NULL	NULL	NULL
24	Jasabilla	NULL	NULL	NULL	NULL	634

Q. Select * from books where author_lname is not null;
Delete
For deleting such null value → ~~Select~~ * from books where title is null;

Q. Select * from book where title is null;
→ empty set

Exercise

Q. Evaluate the following

Select $10! = 10$;

Select $15 > 14$ And $99 - 5 \leq 94$

Select 1 in {5, 3} or 9 between 8 and 10;

Q. Select all books written Before 1900 (Non inclusive)

Q. Select all books written by Eggers or Chabon

Q. Select all books written by Lahiri published after 2000

Q. Select all books with page counts between 100 and 200

Q. Select all books where author_lname starts with a 'c' or an 's'

Q. Using case point title, author_lname, type
if title contains 'Stories' → short stories

Just kids and A Heartbreaking → memoir
work ...

Everything else → Novel

0.8 Bonus generate the output

author-fname	author-lname	count
Thumpa	Lahiri	2 books
Neil	Gaiman	3 books
Dave	Eggws	3 books
Nichol	Chabon	1 book
Patti	Smith	1 book
"	"	"

select author-fname, author-lname

CASE

when count(*) = 1 THEN '1 book'

ELSE CONCAT(COUNT(*), ' books')

END AS count

FROM books

WHERE author-lname is NOT NULL

Group by author-fname, author-lname;

Section-12

UNIQUE CONSTRAINT अपर्याप्त दूसरा नियम यहाँ auto inc कोड

Something unique can't be repeated

Ex. 1 ① CREATE Table contacts (name VARCHAR (100) NOT NULL,
phone VARCHAR(15) NOT NULL UNIQUE);

→ ② insert into contacts (name, phone) value
('Billibob', '87892345')

③ select * from contacts;

name	phone
billybob	87892345

Phone is unique
, name can be duplicated

→ ④ insert into contacts (name, phone) value ('Timmy',
87892345);

Output → Duplicate entry for key 'contacts.phone'

CHECK CONSTRAINTS

CREATE TABLE parties (name VARCHAR (50),
age INT check (age>18);

Age must be greater than 18

insert into partitions (name, age) values ('ramu', 18);
output → check constraint 'usns-chk-1' is violated.
Not a valid syntax

Ex-2 → create table palindromes (word VARCHAR(100)

CHECK (REVERSE(word)=word));

✓ insert into palindromes (word) values ('raucar');

✗ insert ~~momys~~ into →

✗ ('momys');

✓

word	values
raucar	raucar

Output Check constraint

palindrome-chk-1 violated

NAMED CONSTRAINTS

① CREATE TABLE partitions (name VARCHAR(50),
age INT, CONSTRAINT age-over-18 CHECK
(age > 18));

- we can provide a name for the constraint

② insert into partitions (name, age) values ('raga', 2);
output → check constraint 'age-over-18' is violated
looks good 

MULTIPLE COLUMN CONSTRAINTS

```
CREATE TABLE companios (name VARCHAR(255) NOT NULL,  
address VARCHAR(255) NOT NULL'  
CONSTRAINT name-address UNIQUE(name,address)  
);
```

- ① ✓ insert into companies(name, address) values ('saja', 'billanagar');
② ✓ " " ("saja", "randonagar");
③ ✓ " " ("saja", "billanagar");
X output Duplicate entry for key companies.
name-address.

name	address
sujata	bhillanagar
sujata	yondanaga

① insert INTO houses (purchase-price, sale-price) VALUES
(500, 300);

Output → Check constraint 'spcid_get_price' is violated

② insert INTO houses (purchase-price, sale-price) VALUES
(500, 3000);

Output

Purchase price	Sale-price
500	3000

ALTER TABLE

Use alter table to add a new column to an existing table

Syntax → ALTER TABLE Companies ADD COLUMN
| city VARCHAR(25); |

① drop companies;

Field	Type	Null	Key	Default	Extra
name	VARCHAR(255)	NO	PRI	NULL	
address	VARCHAR(255)	NO	PRI	NULL	

② alter table companies add column city VARCHAR(40);
Select * From companies;

name	address	city
Raja	billonagar	NULL
Raja	gandnagar	NULL

* The default value will change for lack of value it is treated as NULL

For example For varchar → Empty string

For Integer → 0

Example-2 → ALTER TABLE Companies

ADD COLUMN employee_count INT NOT NULL;
or ↑ optional

ADD employee_count INT NOT NULL;

name	address	city	Employee-count
..	0 ↳
..	0

~~when~~ we can change 0 to 1 by typing

... INT NOT NULL DEFAULT 1 ;

ALTER TABLE DROPPING COLUMNS

Q. How to remove column from a table?

ALTER TABLE {tablename}

DROP COLUMN {columnname};
↑ optional

→ ALTER TABLE Companies DROP COLUMN
city 2 ;

→ Delete table column

ALTER TABLE RENAMING

Q. How to rename a table ?

RENAME TABLE companies TO suppliess ;

OR

ALTER TABLE suppliess RENAME TO companies ;

Q. How to rename a column ?

Syntax

ALTER TABLE suppliess ← current table name

RENAME COLUMN name TO biz-name ;

Not optional current name ↑ new name
must required

MODIFY

Q. How to modify the column definitions ?

ALTER TABLE table-name MODIFY column-name
 VARCHAR(100) ;

* use modify to change a existing column's type

ALTER TABLE companies MODIFY company-name
 VARCHAR(100) DEFAULT 'Unknown' ;

insert into companies (address) values ('2554 rose lane');
select * from companies

company-name	address
blackbird auto	123 spruce
luigis pies	123 spruce
unknown	2554 rose lane

q. what if I want to rename a column and change its definitions simultaneously ?

→ ALTER TABLE companies CHANGE column-name
^ new-column-name VARCHAR(50);
spau

ALTER TABLE CONSTRAINTS

q. On the houses price what if we want to modify the code such that only positive purchase price is accepted ?
we have coded it previously to modify constraint

Syntax ?

① ALTER TABLE houses ADD CONSTRAINT positive_price
CHECK (purchase-price >= 0);

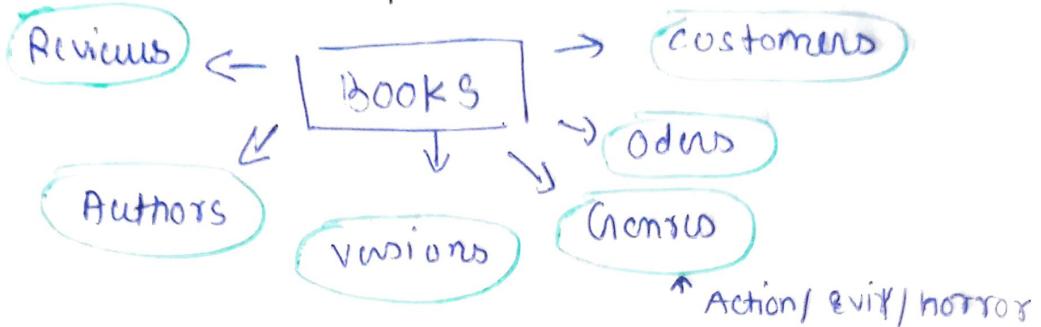
q. now we want to remove what we did earlier ?

ALTER TABLE houses DROP CONSTRAINT positive_price;

Section-13

RELATIONSHIP AND JOINS

For now we are working only with simple data
Now its time for complex data



→ Just when we vision amazon / flipkart we see different categories if we are looking only for books. But we need to think about how are they interrelated 😊

∴ So where do we start ?

RELATIONSHIP BASICS

- (i) one to one relationship.
- (ii) one to many relationship
- (iii) Many to many relationship

Customer \rightleftharpoons Review (one to one)

1 customer can review 1 book that he brought via vnsa

Books \rightleftharpoons Reviews

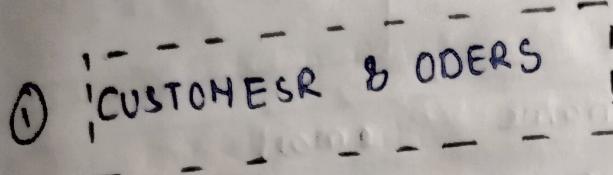
1 to many (Books can be reviewed by many persons)
but one person can review only one book

Books \rightleftharpoons Authors

Many to Many (1 book can have multiple authors
& multiple authors can have multiple books)

1 : MANY

The most common relationship



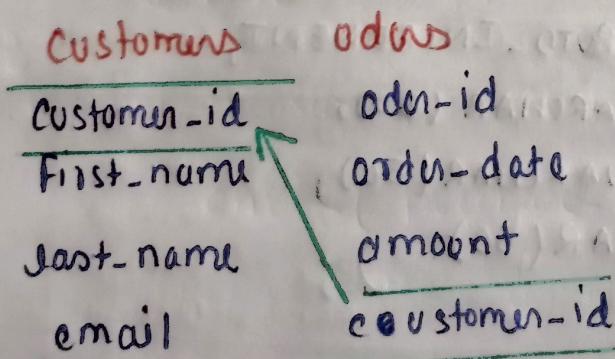
what we want to store?

A customer first and last name, A customer's email, The date of the purchase, The price of the order

We could use one table

first-name	last-name	email	order-date	amount
Boy	George	george@gmail.com	2016-02-10	99.99
Boy	George	george@gmail.com	2017/11/11	35.30
George	Michel	gm@gmail.com	2014-12-12	800.67
George	Michel	gm@gmail.com	2015/01/03	12.50
David	Browne	david@gmail.com	NULL	NULL
Bill	Stall	blue@gmail.com	NULL	NULL

But not a good idea



Foreign key can refer to a given table within a table

PRIMARY KEY

CUSTOMERS

Customer-id	first-name	last-name	email
1	Boy	George	boy@newupdate.com
2	George	Michel	george@random.com
3	David	Bowie	bowie@slowing.it
4	Blue	Stall	blue@newupdate.com

ORDERS

FOREIGN KEY

order-id	order-date	amount	customer-id
1	'2016/02/10'	99.99	1
2	..	35.50	2
3	..	800.67	2
4	..	12.50	2

CREATE TABLE customers (id INT

unique value

← **PRIMARY KEY** AUTO_INCREMENT,
 first-name VARCHAR(50),
 last-name VARCHAR(50),
 email VARCHAR(50));

CREATE TABLE ORDERS (

id INT PRIMARY KEY AUTO_INCREMENT,
 order-date DATE,
 amount DECIMAL(8,2)
 customer-id INT,

FOREIGN KEY (customer-id) REFERENCES customers(id)

);

- ① INSERT INTO " "
- ② INSERT INTO ORDERS . . .

Now, if we are inserting any values whose customer-id doesn't exist in customers table we will get an error

insert into orders(order-date, amount, customer-id) values
('2022-11-11', 78.00, 42);

it will fail

output → cannot add/update a child row... a foreign key constraint fails . . . ✓

CROSS JOINS

- Q. How will you find the order details of a person last-name 'George' ;

- ① Select id from customers WHERE last-name = 'George';
~~Forgot the id~~ ↗ output = 1
- ② Select * from orders where customer-id = 1;

OR (using subquery)

Select * from orders where customer-id = (SELECT id FROM customers WHERE last-name = 'George');

but we are not getting customers table, it is good to have the information joined together

Cross join will take every single row from customers and join it with every single row from orders.

Select * From customers, orders;

जब जाना चाहे उसका लिया जाए है। OR orders, customers

वर्ताव से है उसका

प्रोड के पुरा

orders के जांच तक
होता है।

id	first-name	last-name	email
1	Boy	George	george@gmail.com
2	George	Michel	gm@gmail.com
3	David	Bowie	david@gmail.com
4	Blue	stall	blue@gmail.com
5	Bette	Davis	bette@aol.com

orders

id	order-date	amount	customer-id
1	2016-02-10	99.99	1
2	2017-11-11	35.50	1
3	2014-12-12	800.67	2
4	2015-01-03	12.50	2
5	1999-04-11	450.25	5

id	order-date	amount	customer-id	id	first-name	last-name	email
5	1999-04-11	450.25	5	1	Boy	George	..
4	2015-01-03	12.50	2	1	Boy	George	..
3	2014-12-12	800.67	2	1	Boy	George	..
2	2017-11-11	35.50	1	1	Boy	George	..
1	2016-02-10	99.99	1	1	Boy	George	..
5	1999-04-11	450.25	5	2	George	Michel	..
4	2015-01-03	12.50	2	2	George	Michel	..
3	2014-12-12	800.67	2	2	George	Michel	..
2	2017-11-11	35.50	1	2	George	Michel	..
1	2016-02-10	99.99	1	2	George	Michel	..

INNER JOIN

Select all records from A and B where the join conditions is met



Syntax → Select * FROM customers

JOIN orders

ON customers.id = orders.customer-id;

JOIN / INNER JOIN

id	first-name	last-name	email	id	order-date	amount	customer-id
1	Boy	George	...	1	2016-02-10	99.99	1
1	Boy	George	...	2	2017-11-11	35.50	1
2	George	Michel	...	3	2014-12-12	800.67	2
2	George	Michel	...	4	2015-01-03	12.50	2
5	Bette	Davis	...	5	1999-04-11	450.25	5

- ② Select first-name, last-name, order-date, amount from customers JOIN orders ON customers.id = orders.customer-id;

first-name	last-name	order-date	amount
Boy	George	2016-02-10	99.99
Boy	George	2017-11-11	35.50
George	Michel	2014-12-12	800.67
George	Michel	2015-01-03	12.50
Bette	Davis	1999-04-11	450.25

③ Select * From orders Join customers on customers.id = orders.id;

Sequence matters

id	order-date	amount	customer-id	id	first-name	last-name	email
1	2016-02-10	99.99	1	1	Boy	George	??
2	2017-11-11	35.50	1	2	George	Michel	??
3	2014-12-22	800.67	2	3	David	Bowie	??
4	2015-01-09	12.50	2	4	Blue	Stule	??
5	1994-04-11	450.25	5	5	Butte	Davis	??

INNER JOINS WITH GROUP BY

Q. How will you find the total amount a unique customer spent?

Select first-name, last-name, sum(amount) FROM CUSTOMERS
 JOIN ORDERS ON orders.customer-id = customers.id
 GROUP BY first-name, last-name;

first-name	last-name	total
Boy	George	135.49
Butte	Davis	450.25
George	Michel	813.17

LEFT JOIN



→ Select everything FROM A along with any matching records from B.

एक type का join है जिसके one side से GET information करता है और दूसरी side की overlapping information का भी।

Ex) Select first-name, last-name, order-date, amount FROM
customers LEFT INNER JOIN orders ON orders.customer-id =
 customers.id ;

first-name	last-name	order-date	amount
Boy	George	2017-11-11	35.50
Boy	George	2016-02-10	99.99
George	Michel	2015-01-03	12.50
George	Michel	2014-12-12	800.67
David	Bowie	NULL	NULL
Blue	Stall	NULL	NULL
Butte	Davis	1999-04-11	450.25

Compare & Check it with normal join table 😊

Q. Why the need of left Join? If we already have normal Join

Ans.) Suppose we want to find the user who didn't purchase anything yet, how then? Normal Join will give us that user info. Therefore left Join is used 😊

Chile-2

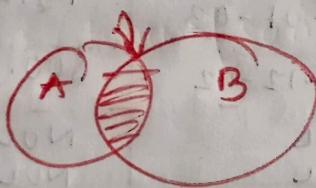
Select ordn-date, amount, first-name, last-name

FROM Orders LEFT JOIN customers ON orders.customer_id = customers.id;

first-name	last-name	ordn-date	amount
Boy	George	2016-02-10	99.99
Boy	George	2017-11-11	35.50
George	Michel	2014-12-12	800.67
George	Michel	2015-01-03	12.50
Bette	Davis	1999-04-11	450.25

Compare the table using JOIN [both are same 😢 how?]

this type every order has a value associated with a customer id this has become a case of inner join.



(Left Join as Outer Join)

LEFT JOIN WITH GROUP BY

SELECT first-name, last-name, sum(amount) ① replaced
IFNULL(SUM(amount), 0) ② AS money-spent
replacing value

FROM customers
 LEFT JOIN
 Orders ON customers.id = orders.customer_id

GROUP BY First-name, Last-name;

No
Space