PW SKILLS

⇒ Java => Sql => JDBC => Servlet + JSP

Hibernate => spring -> MongoDB -> Js & React

⇓

Microservices

=> Sql => Structured Query Language

=> Data => Info

=> Rohan -> info -> name

=> sachin
50
Mumbai
} Data of sachin

=> DataBase -> large volume of info / Data

└> DB :- Every one -> own way of maintaining Data

=> Ledger 

=> computer => store info => files -text files

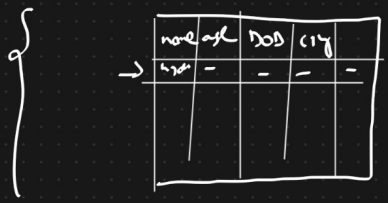
=> excell -1

=> Software => store Data (large)

DBMS => sybase
Informix
Microsoft
Oracle

DBMS :- store Data Rows & Columns (Table)

⇓



Relational DBMS

=> Person -> info -> Multiple tables

=>

=> RDBMS -> Tables -> rows & clums => Multiple table

=> MySQL
=> Postgral
-> Oracle Small
MariaDB
SQLite
:

} SQL

98 -> sql same
2% varies

NoSQL (key-value, Docum, graph ---)

=> MongoDB - -
=> Hbase -
=> Cassandra - -
:

X= sql

=> RDBMS :-

Oracle =>

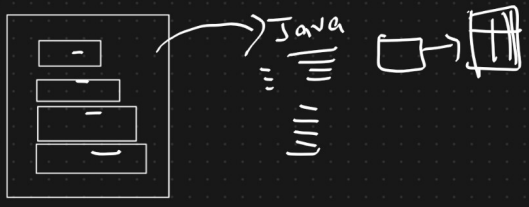
MySQL =>

=> MySQL :- SQL -> Structured Query Language

=> (1) Straight forward }

CRUD -> Query write => suffient

{MVC}



=> MySQL => installation

↳ NeuroLabs => MySQL → Guide → Doit

MySQL → workbench & server

=> Create DataBase ;

SQL -> standard

↓ ↳ commands → capital letter

case insensitive

=> DataBase → creation

↳ use Database name

=> create table

=> Perform => CRUD => operations

=> Creation of DataBase =>

Create Database iNeuron; INEURON

=> Table => rows & colums

↳ use iNeuron; ↳

Create table tablename (____);

=> CREATE => Database

=> Table

=> => CREATE TABLE student (-----); ——

create table student (- - -); ——

=> Data Base

⇒ use iNeuron ;

numeric ⇒ int
String ⇒ varchar

⇒ CREATE TABLE student (

```
(
    id int ,
    name varchar ,
    age int ,
    city varchar
);
```

(256) <=> id int(5); 12145
name varchar(50),
age int(5),
city varchar(25)
);

⇒ show Databases ; SHOW DATABASES ;

show tables ;

⇒ Describe table ;

⇒ Desc table ;

⇒ CRUD operation } After DB creation & table creation

C → insert statements
R → select statements
U → update statements
D → delete statements