https://archive.ics.uci.edu/ml/machine-learning-databases/00222/
https://archive.ics.uci.edu/ml/datasets/bank+marketing show databases create database if not exists ineuron use ineuron create table bank_details (age int , job varchar(30) , marital varchar(30) , education varchar(30), `default` varchar(30), balance int, housing varchar(30) , loan varchar(30) , contact varchar(30) , `day` int , `month` varchar(30) , duration int , campaign int , pdays int , previous int , poutcome varchar(30), y varchar(30)) create table if not exists bank_details (age int , job varchar(30) , marital varchar(30) , education varchar(30), `default` varchar(30), balance int, housing varchar(30) , loan varchar(30) , contact varchar(30) , `day` int , `month` varchar(30) , duration int , campaign int , pdays int , previous int , poutcome varchar(30), y varchar(30)) show tables Describe bank_details insert into bank_details values
(58,"management","married","tertiary","no",2143,"yes","no","unknown",5,"may",261,1,-1,0,"unknown","no") insert into bank_details values
(44,"technician","single","secondary","no",29,"yes","no","unknown",5,"may",151,1,-1,0,"unknown","no"),
(33,"entrepreneur","married","secondary","no",2,"yes","yes","unknown",5,"may",76,1,-1,0,"unknown","no"),
(47,"blue-collar","married","unknown","no",1506,"yes","no","unknown",5,"may",92,1,-1,0,"unknown","no"),
(33,"unknown","single","unknown","no",1,"no","no","unknown",5,"may",198,1,-1,0,"unknown","no"),
(35,"management","married","tertiary","no",231,"yes","no","unknown",5,"may",139,1,-1,0,"unknown","no"),
(28,"management","single","tertiary","no",447,"yes","yes","unknown",5,"may",217,1,-1,0,"unknown","no"),
(42,"entrepreneur","divorced","tertiary","yes",2,"yes","no","unknown",5,"may",380,1,-1,0,"unknown","no"),
(58,"retired","married","primary","no",121,"yes","no","unknown",5,"may",50,1,-1,0,"unknown","no"),
(43,"technician","single","secondary","no",593,"yes","no","unknown",5,"may",55,1,-1,0,"unknown","no"),
(41,"admin.","divorced","secondary","no",270,"yes","no","unknown",5,"may",222,1,-1,0,"unknown","no"),
(29,"admin.","single","secondary","no",390,"yes","no","unknown",5,"may",137,1,-1,0,"unknown","no"),
(53,"technician","married","secondary","no",6,"yes","no","unknown",5,"may",517,1,-1,0,"unknown","no"),
(58,"technician","married","unknown","no",71,"yes","no","unknown",5,"may",71,1,-1,0,"unknown","no"),
(57,"services","married","secondary","no",162,"yes","no","unknown",5,"may",174,1,-1,0,"unknown","no"),
(51,"retired","married","primary","no",229,"yes","no","unknown",5,"may",353,1,-1,0,"unknown","no"),
(45,"admin.","single","unknown","no",13,"yes","no","unknown",5,"may",98,1,-1,0,"unknown","no"),
(57,"blue-collar","married","primary","no",52,"yes","no","unknown",5,"may",38,1,-1,0,"unknown","no"),
(60,"retired","married","primary","no",60,"yes","no","unknown",5,"may",219,1,-1,0,"unknown","no") select * from bank_details select age , job from bank_details select `default` , age from bank_details select * from bank_details where age = 41 select job from bank_details where age =41 select * from bank_details where job = 'retired' and balance > 100 select * from bank_details where education = 'primary' or balance < 100 select * from bank_details where education = 'primary' and balance < 100 select * from bank_details select age , job from bank_details select `default` , age from bank_details select * from bank_details where age = 41 select job from bank_details where age =41 select * from bank_details where job = 'retired' and balance > 100 select * from bank_details where education = 'primary' or balance < 100 select * from bank_details where education = 'primary' and balance < 100 select distinct job from bank_details select * from bank_details order by age select * from bank_details order by age desc select count(*) from bank_details select sum(balance) from bank_details select avg(balance) from bank_details select * from bank_details where balance = (select min(balance) from bank_details) set sql_safe_updates = 0 update bank_details set balance = 0 where job = 'unknown' update bank_details set contact = 'known' , y = 'yes' where month = 'may' update bank_details set `default` = 'NULL' where `default`= 'no'; DELIMITER && create procedure select_pre() BEGIN select * from bank_details; END && call select_pre() DELIMITER && create procedure select_pre_filter() BEGIN select * from bank_details where job = 'retired' and balance > 100; END && call select_pre_filter() DELIMITER && create procedure select_pre_filter2(IN var int ,IN var1 varchar(30)) BEGIN select * from bank_details where job = var1 and balance > var; END && call select_pre_filter2(100 ,'services') call select_pre() select * from (select job , age , education , y from bank_details) as a where a.age = 58 select job , age , education, y from bank_details where age = 58 create view bank_view as select job , age , education , y from bank_details select * from bank_view where age = 58 create table if not exists bank_details1 (age int , job varchar(30) , marital varchar(30) , education varchar(30), `default` varchar(30), balance int, housing varchar(30) , loan varchar(30) , contact varchar(30) , `day` int , `month` varchar(30) , duration int , campaign int , pdays int , previous int , poutcome varchar(30), y varchar(30)) show tables insert into bank_details1 select * from bank_details ; select * from bank_details1 create table if not exists bank_details2 (age int , job varchar(30) , marital varchar(30) , education varchar(30), `default` varchar(30), balance int, housing varchar(30) , loan varchar(30) , contact varchar(30) , `day` int , `month` varchar(30) , duration int , campaign int , pdays int , previous int , poutcome varchar(30), y varchar(30)) show tables insert into bank_details2 select * from bank_details where age = 58; select * from bank_details2 select bank_details.age , bank_details.job , bank_details.marital from bank_details inner join bank_details2 on bank_details.age =bank_details2.age select * from bank_details select

bank_details.age , bank_details.job , bank_details.marital from bank_details right join bank_details2 on bank_details.age =bank_details2.age select bank_details.age , bank_details.job , bank_details.marital from bank_details left join bank_details2 on bank_details.age =bank_details2.age https://cloud.mongodb.com/v2/61067239fcc7ab33b6d8477e#clusters pip install "pymongo[srv]" client = pymongo.MongoClient("mongodb+srv://ineuron:mongodb123@cluster0.goi2j.mongodb.net/?retryWrites=true&w=majority") db = client.test print(db) import pymongo client = pymongo.MongoClient("mongodb+srv://ineuron:mongodb123@cluster0.goi2j.mongodb.net/?retryWrites=true&w=majority") db = client.test print(db) d = { "name":"sudhanshu", "email" : "sudhanshu@ineuron.ai", "surname" : "kumar" } db1 = client['mongotest'] coll = db1['test'] coll.insert_one(d) git https://git-scm.com/downloads https://github.com/sudh9931/testgitpush.git