**Lab 01**

**mysql -u root -p**

**show databases;**

**create database** THK\_Database;

**use** thk\_database;

**create table** lab\_grades

(

std\_id char(4),

name varchar(30),

major char(3),

section char(1),

days\_present int,

project\_marks double,

cgpa decimal(3,2),

submission\_date date

);

**show tables;**

**describe** lab\_grades;

**insert into** lab\_grades **values** ('s001', 'Rafa', 'CSE', '1', '10', '18.5', '3.91', '28-01-24');

**select \* from** lab\_grades;

**select** name, cgpa **from** lab\_grades;

**alter table** lab\_grades **add** project\_title char(10);

**alter table** lab\_grades **modify column** project\_title varchar(50);

**alter table** lab\_grades **drop column** project\_title;

**alter table** lab\_grades **rename column** submission\_date **to** sub\_date;

**alter table** lab\_grades **change** submission\_date sub\_date date;

**update** lab\_grades **set** name = 'Aria' **where** std\_id = 's001';

**delete from** lab\_grades **where** sub\_date = '2028-01-24';

**select** std\_id, project\_marks **from** lab\_grades;

**select** name, project\_marks+days\_present\*5/12 **as** Total\_marks **from** lab\_grades;

**select** upper(name), Lower(name) **from** lab\_grades;

**select distinct** major **from** lab\_grades;

**select \* from** lab\_grades **order by** cgpa desc;

**select \* from** lab\_grades **order by** cgpa **desc**, sub\_date;

**select** name,project\_marks **from** lab\_grades **where** major **=** 'cse';

**select** name,project\_marks **from** lab\_grades **where** project\_marks **between** 17 **and** 19;

**select \* from** lab\_grades **where** major **in** ('cse', 'cs');

**select \* from** lab\_grades **where** major **=** 'cse' **or** 'cs';

**select \* from** lab\_grades **where** project\_marks>18 **and** sub\_date **between** '2018-08-01' **and** '2018-08-31';

**select \* from** lab\_grades **where** name **like** 'a**%**';

**select \* from** lab\_grades **where** name **like** 'a**\_\_\_**';

**Lab 02**

**Select** min(cgpa) **from** lab\_grades;

**Select** count(\*) **as** total\_studs, avg(project\_marks) **as** Average **from** lab\_grades;

**Select sum**(days\_present) **from** lab\_grades;

**Select max**(submission\_date) **from** lab\_grades;

**Select** major, **min**(cgpa), **max**(cgpa) **from** lab\_grades **group by** major;

**Select** major, **count**(\*) **from** lab\_grades **group by** major;

**Select** major, **min**(cgpa), **max**(cgpa) **from** lab\_grades **group by** major **having count**(\*) >=2;

**Select** major, **min**(project\_marks), **max**(project\_marks) **from** lab\_grades **where** submission\_date <= '2018-09-15'  **group by** major;

**Select** name from lab\_grades **where** project\_marks =(**Select** **max**(project\_marks) **from** lab\_grades);

**Select \* from** lab\_grades **where** major = 'cse' **and** cgpa> **any**(**select** cgpa **from** lab\_grades **where** major **=** 'cs');

**Select \* from** lab\_grades **where** major **=** 'cse' **and** cgpa> **all**(**select** cgpa **from** lab\_grades **where** major **=** 'cs');

**Select distinct** major **from** lab\_grades l1 **where exists** (**select \* from** lab\_grades l2 **where l2.major = l1.majo**r **and** l2.cgpa<2.7);

**Select** name **from** lab\_grades l1 w**here exists** (**select \* from** lab\_grades l2 **where** l2.major = l1.major and l1.project\_marks > l2.project\_marks );

**Select** name **from** lab\_grades l1 **where not exists (select \* from** lab\_grades l2 **where** l2.std\_id **!=** l1.std\_id **and** l2.project\_marks > l1.project\_marks);

**Select count(\*) from** lab\_grades l1 **where not exists** (**select \* from** lab\_grades l2 **where** l2.std\_id **!=** l1.std\_id **and** l2.project\_marks > l1.project\_marks );

**Select** major **from** lab\_grades **group by** major **having count**(\*) >= **all**(**Select** **count**(\*) **from**

lab\_grades **group by** major);

**Lab 03**

**CREATE DATABASE** Bank;

**USE** Bank;

**create table** customer (

customer\_id varchar(10) **not null**,

customer\_name varchar(20) **not null**,

customer\_street varchar(30),

customer\_city varchar(30),

**primary key (customer\_id)**);

**create table** branch (

branch\_name varchar(15),

branch\_city varchar(30),

assets int,

**primary key (branch\_name)**,

**check (assets >= 0)**);

**create table** account (

branch\_name varchar(15),

account\_number varchar(10) **not null**,

balance int,

**primary key (account\_number)**,

**check (balance >= 0)**);

**create table** loan (

loan\_number varchar(10) **not null**,

branch\_name varchar(15),

amount int,

**primary key (loan\_number)**);

**create table** depositor (

customer\_id varchar(10) **not null**,

account\_number varchar(10) **not null**,

**primary key (customer\_id,account\_number)**,

**foreign key (customer\_id) references customer(customer\_id)**,

**foreign key (account\_number) references account(account\_number)**);

**create table** borrower (

customer\_id varchar(10) **not null**,

loan\_number varchar(10) **not null**,

**primary key (customer\_id, loan\_number)**,

**foreign key (customer\_id) references customer(customer\_id)**,

**foreign key (loan\_number) references loan(loan\_number)**);

**insert into** customer **values**

('C-101','Jones', 'Main', 'Harrison'),

('C-201','Smith', 'North', 'Rye'),

('C-211','Hayes', 'Main', 'Harrison'),

('C-212','Curry', 'North', 'Rye'),

('C-215','Lindsay', 'Park', 'Pittsfield'),

('C-220','Turner', 'Putnam', 'Stamford'),

('C-222','Williams', 'Nassau', 'Princeton'),

('C-225','Adams', 'Spring', 'Pittsfield'),

('C-226','Johnson', 'Alma', 'Palo Alto'),

('C-233','Glenn', 'Sand Hill', 'Woodside'),

('C-234','Brooks', 'Senator', 'Brooklyn'),

('C-255','Green', 'Walnut', 'Stamford');

**insert into** branch **values**

('Downtown', 'Brooklyn',9000000),

('Redwood', 'Palo Alto',2100000),

('Perryridge', 'Horseneck',1700000),

('Mianus', 'Horseneck',400000),

('Round Hill', 'Horseneck',8000000),

('Pownal', 'Bennington',300000),

('North Town', 'Rye',3700000),

('Brighton', 'Brooklyn',7100000);

**insert into** account **values**

('Downtown','A-101',500),

('Mianus','A-215',700) ,

('Perryridge','A-102',400),

('Round Hill','A-305',350),

('Brighton','A-201',900),

('Redwood','A-222',700),

('Brighton','A-217',750);

**insert into** loan **values**

('L-17', 'Downtown', 1000),

('L-23', 'Redwood', 2000),

('L-15', 'Perryridge', 1500),

('L-14', 'Downtown', 1500),

('L-93', 'Mianus', 500),

('L-11', 'Round Hill', 900),

('L-16', 'Perryridge', 1300);

**insert into** depositor **values**

('C-226', 'A-101'),

('C-201', 'A-215'),

('C-211', 'A-102'),

('C-220', 'A-305'),

('C-226', 'A-201'),

('C-101', 'A-217'),

('C-215', 'A-222');

**insert into** borrower **values**

('C-101', 'L-17'),

('C-201', 'L-23'),

('C-211', 'L-15'),

('C-226', 'L-14'),

('C-212', 'L-93'),

('C-201', 'L-11'),

('C-222', 'L-17'),

('C-225', 'L-16');

**Select** customer**.**customer\_id, customer\_name, customer\_city, account\_number **from** customer **inner join** depositor **on** customer**.**customer\_id **=** depositor**.**customer\_id**;**

**Select** customer**.**customer\_id, customer\_name, customer\_city, account\_number **from** customer **left join** depositor **on** customer**.**customer\_id **=** depositor**.**customer\_id**;**

**Select** customer**.**customer\_id, customer\_name, customer\_city, account\_number **from** customer **right join** depositor **on** customer**.**customer\_id **=** depositor**.**customer\_id**;**

**Select** customer**.**customer\_id, customer\_name, customer\_city, account\_number **from** customer **outer join** depositor **on** customer**.**customer\_id **=** depositor**.**customer\_id**;**

**Select** customer.customer\_name, customer\_city, depositor.account\_number, balance, branch\_name **from ((**customer **inner join** depositor **on** customer**.**customer\_id **=** depositor**.**customer\_id**) inner join** Account **on** account.account\_number **=** depositor.account\_number**);**

**Select** customer\_name, customer\_city, depositor.account\_number, balance, branch\_name **from** customer, depositor, Account **where** customer**.**customer\_id **=** depositor**.**customer\_id **and** depositor.account\_number **=** account.account\_number**;**

**Select** customer\_name, customer\_city **from** ((customer **inner join** borrower **on** customer.customer\_id **=** borrower.customer\_id) **inner join** loan **on** borrower.loan\_number **=** loan.loan\_number) **where** loan.branch\_name **=** 'Perryridge';

**Select** account\_number **from** account **where** balance **between** 700 **and** 900**;**

**Select** customer\_name **from** customer **where** customer\_street **like** '**%**hill';

**Select** branch\_name **from** branch **where** assets > **any**(**select** assets **from** branch **where** branch\_city **=** 'Brooklyn');

**Select** branch\_name **from** branch **where** assets > **all**(**select** assets **from** branch **where** branch\_city **=** 'Horseneck');

SELECT customer\_name FROM ((customer Inner JOIN depositor ON customer.customer\_id = depositor.customer\_id) Inner JOIN account ON depositor.account\_number = account.account\_number) WHERE account.branch\_name = 'Brighton' ORDER BY customer\_name;

SELECT \* FROM loan ORDER BY amount DESC, loan\_number ASC;

SELECT branch\_name FROM account GROUP BY branch\_name HAVING avg(balance) >= 700;

SELECT customer\_name, account\_number FROM ((customer Inner JOIN depositor ON customer.customer\_id = depositor .customer\_id) Inner JOIN account ON depositor.account\_number = account.account\_number) ORDER BY account.balance DESC

LIMIT 3;

SELECT DISTINCT customer.customer\_name FROM ((customer Inner JOIN depositor ON customer.customer\_id = depositor.customer\_id) Inner JOIN account ON depositor.account\_number = account .account\_number) WHERE account.branch\_name IN (

SELECT branch\_name FROM ((depositor Inner JOIN account a ON depositor.account\_number = account .account\_number) Inner JOIN customer ON depositor .customer\_id = customer.customer\_id) WHERE customer.customer\_name = 'Johnson');

Select distinct customer\_name from ((((customer inner join depositor on customer .customer\_id = depositor.customer\_id) inner join account ON depositor.account\_number = account.account\_number) left join borrower on customer.customer\_id = borrower.customer\_id)

Left join loan on borrower.loan\_number = loan.loan\_number and loan.branch\_name = 'Mianus')where account.branch\_name = 'Mianus' and borrower.loan\_number is null;

SELECT branch.branch\_name, COUNT(DISTINCT depositor.customer\_id) AS num\_customers

FROM branch

LEFT JOIN account ON branch.branch\_name =account.branch\_name

LEFT JOIN depositor ON account.account\_number = depositor.account\_number

GROUP BY branch.branch\_name;

SELECT AVG(balance) AS average\_balance

FROM (

SELECT c.customer\_id, COUNT(a.account\_number) AS num\_accounts, AVG(a.balance) AS balance

FROM customer c

JOIN depositor d ON c.customer\_id = d.customer\_id

JOIN account a ON d.account\_number = a.account\_number

WHERE c.customer\_city = 'Palo Alto'

GROUP BY c.customer\_id

HAVING num\_accounts >= 2

) AS subquery;

SELECT c.customer\_name, a.account\_number, a.balance

FROM customer c

JOIN depositor d ON c.customer\_id = d.customer\_id

JOIN account a ON d.account\_number = a.account\_number

ORDER BY a.balance DESC

LIMIT 1 OFFSET 2;