Subject: Informatics Practices (065)
Topic: MySQL Queries Part-III
Subject Instructor: Mr. Prabhat Das (PGT-IP) | Army Public School Jorhat
Email: prabhat@apsjorhat.org

Tables:

emp_details

Sl_No	EMP_NAME	SALARY	DOB	PHONE	DEPT	bank_acc_no
101	SAM	55000	2004-10-10	911	IT	101
202	RAM	33000	2006-10-12	100	SECURITY	202
303	BOB	88000	2003-10-10	102	IT	303
404	CHRIS	110000	2003-01-10	500	IT	101
505	ANGELA	132000	2000-11-21	999	PURHCASE	303
606	ALEX	220000	1999-10-10	777	NULL	303
707	SUZUKA	109999	2000-10-10	72	IT	303

bank_details

account_num	bank_name
101	STATE BANK OF INDIA
202	PUNJAB NATIONAL BANK
303	BANK OF BARODA

Q1: Write a query to display the sum, avg, max and min of emp_salary from emp_details table.

Sol: select sum(emp_salary), avg(emp_salary), max(emp_salary), min(emp_salary) from emp_details;

Q2: Write a query to display the number of employees with same department.

Sol: select dept, count(*) from emp_details group by dept;

Q3: Write a query to display the highest salary of an employee from each department.

Sol: select max(emp_salary) from emp_details group by dept;

Q4: Write a query to display the total salary given to the employees of a particular department having column name as expenditure.

Sol: select dept, sum(emp_salary) expenditure from emp_details group by dept;

Q5: Display the details of employee with highest salary.

Sol: select * from emp_details order by emp_salary desc limit 1;

Q6: Write a query to display the total number of employees in each department with total number of employees > 2 and Salary > 10,000.

Sol: select dept, count(*) number_of_employees from emp_details where emp_salary > 10000 group by dept having number_of_employees > 2;

Q7: Write a query in MySQL to get the bank_name and emp_name from emp_details and bank_details table.

Sol: select emp_details.emp_name, bank_details.bank_name from emp_details, bank_details where emp_details.bank_acc_no = bank_details.account_num;

Q8: Write a query to display the details of all employees in descending order of the emp_salary.

Sol: select * from emp_details order by emp_salary desc;

Q9: Write a query to display the distinct number of departments in emp_details table.

Sol: select dept, count(*) from emp_details group by dept;
Or, select dept from emp_details group by dept;

Q10: Write the output of the following:

select dept, max(emp_salary), count(*) from emp_details group by dept;

Sol:

dept	max(emp_salary)	count(*)
NULL	220000	2
SECURITY	33000	1
IT	110000	3
PURCHASE	132000	1

Q11: Create book_info table, sales table & exam_results table with the columns given below and insert five values.

book_info

Column Name
BOOK_ID
BOOK_TITLE
PRICE

exam_results

Column Name
STU_ID
FNAME
LNAME
EXAM_ID
EXAM_SCORE

sales

COLUMN NAME
STORE_ID
SALES_DATE
SALES_AMOUNT

Sol:
Creating Tables:

create table book_info(BOOK_ID int(15), BOOK_TITLE varchar(50), PRICE int(8));

create table sales(STORE_ID int(15), SALES_DATA varchar(30), SALES_AMOUNT int(8));

create table exam_results(STU_ID int(5), FNAME varchar(10), LNAME varchar(10), EXAM_ID int(2), EXAM_SCORE int(5));

Inserting values:

insert into book_info values(101, "TO KILL A MOCKING BIRD", 349), (202, "HARRY POTTER AND THE PHILOSOPHER'S STONE", 999), (303, "A PASSAGE TO INDIA", 599), (404, "INVISIBLE MAN", 249), (505, "THE ALCHEMIST", 549);

insert into sales values(101, "APPLE INC.", 999999), (202, "FACEBOOK INC.", 777777), (303, "SPACE X", 888888), (404, "TESLA", 899999), (505, "TATA", 799999);

insert into exam_results values(101, "BORIS", "JOHNSON", 1, 90), (101, "BORIS", "JOHNSON", 2, 85), (202, "JONATHA", "JOESTAR", 1, 99), (202, "JONATHA", "JOESTAR", 2, 75), (303, "JOHNNY", "HARRIS", 1, 100), (303, "JOHNNY", "HARRIS", 2, 95), (404, "WILLIAM", "CROOKS", 1, 88), (404, "WILLIAM", "CROOKS", 2, 90), (505, "JOHN", "FRANCS", 1, 65), (505, "JOHN", "FRANCS", 2, 45);

Q12: Which SQL statement allows you to find the highest price from the table book_info?

Sol: select max(PRICE) from book_info;

Q13: Which SQL statement lets you find the SALES amount for each store?

Sol: select STORE_ID, SUM(SALES_AMOUNT) from SALES group by STORE_ID;

Q14: Which SQL statement lets you list all stores whose total sales amount is over 5000?

Sol: select STORE_ID, SUM(SALES_AMOUNT) from SALES group by STORE_ID having sum(SALES_AMOUNT) > 5000;

Q15: Which SQL statement lets you find the total number of stores in the SALES table?

Sol: select count(STORE_ID) from SALES;

Q16: Which SQL statement allows you to find the total sales amount for STORE_ID 303 and the total sales amount for STORE_ID 505?

Sol: select store_id, sales_data, sum(sales_amount) from sales group by store_id having store_id in (303, 505);

Q17: What SQL statement do we use to find the average exam score from EXAM_ID = 1?

Sol: select avg(EXAM_SCORE) from EXAM_RESULTS group by EXAM_ID having EXAM_ID = 1;

Q18: Which SQL statement do we use to find how many students took each exam?

Sol: select EXAM_ID, count(DISTINCT STU_ID) from EXAM_RESULTS group by EXAM_ID;

Q19: What SQL statement do we use to print out the record of all students whose last name starts with 'L'?

Sol: select * from EXAM_RESULTS where LNAME like 'L%';

Q20: What is the result of the following SQL statement?
select max(EXAM_SCORE) from EXAM_RESULTS group by EXAM_ID having EXAM_ID = 1;

Sol: 90

Q21: Give the output of the following SQL statements:

Table: CLUB

COACH_ID	COACHNAME	AGE	SPORTS	DATOFAPP	PAY	SEX
1	KUKREJA	35	KARATE	27/03/1996	1000	M
2	RAVINA	34	KARATE	20/01/1998	1200	F
3	KARAN	34	SQUASH	19/02/1998	2000	M
4	TARUN	33	BASKETBALL	01/01/1998	1500	M
5	ZUBIN	36	SWIMMING	12/01/1998	750	M
6	KETAKI	36	SWIMMING	24/02/1998	800	F
7	ANKITA	39	SQUASH	20/02/1998	2200	F
8	ZAREEN	37	KARATE	22/02/1998	1100	F
9	KUSH	41	SWIMMING	13/01/1998	900	M
10	SHAILYA	37	BASKETBALL	19/02/1998	1700	M

i) SELECT COUNT(DISTINCT SPORTS) FROM CLUB:

Sol:

COUNT(DISTINCT SPORTS)
4

ii) SELECT MIN(AGE) FROM CLUB WHERE SEX = “f”;

Sol:

MIN(AGE)
34

iii) SELECT AVG(PAY) FROM CLUB WHERE SPORTS = “KARATE”;

Sol:

AVG(PAY)
1033.3333

iv) SELECT SUM(PAY) FROM CLUB WHERE DATOFAPP > “30/01/98”;

Sol:

SUM(PAY)
NULL

Q22: Given the following table:

Table: STUDENT

No.	Name	Stipend	Stream	AvgMark	Grade	Class
1	Karan	400.00	Medical	78.5	B	12B
2	Divakar	450.00	Commerce	89.2	A	11C
3	Divya	300.00	Commerce	68.6	C	12C
4	Arun	350	Humanities	73.1	B	12C
5	Sabina	500.00	Nonmedical	90.6	A	11A
6	John	400.00	Medical	75.4	B	12B
7	Robert	250.00	Humanities	64.4	C	11A
8	Rubina	450.00	Nonmedical	88.5	A	12A
9	Vikas	500.00	Nonmedical	92.0	A	12A
10	Mohan	300.00	Commerce	67.5	C	12C

Give the output of following SQL statements:

i) SELECT MIN(AvgMark) FROM STUDENT WHERE AvgMark < 75;

Sol:

MIN(AvgMark)
64.4

ii) SELECT SUM(Stipend) FROM Student WHERE Grade = 'B';

Sol:

SUM(Stipend)
1150

iii) SELECT AVG(Stipend) FROM Student WHERE CLASS = '12A';

Sol:

AVG(Stipend)
475

iv) SELECT COUNT(DISTINCT) FROM Student;
Sol: *error*

Q23: In a Database, there are two tables given below:

EMPLOYEEID	NAME	SALES	JOBID
E1	SUMIT SINHA	1100000	102
E2	VIJAY SINGH TOMAR	1300000	101
E3	AJAY RAJPAL	1400000	103
E4	MOHIT RAMNANI	1250000	102
E5	SAHAILJA SINGH	1450000	103

JOBID	JOBTITLE	SALARY
101	President	200000
102	Vice President	125000
103	Administration Assistant	80000
104	Accounting Manager	70000
105	Accountant	65000
106	Sales Manager	80000

Write SQl Queries for the following:

i) To display employee ids, names of employees, job ids with corresponding job titles.

Sol: select EMPLOYEEID, NAME, JOBTITLE from employee, job where employee.jobid = job.jobid;

ii) To display names of employees, sales and corresponding job titles who have achieved sales more than 1300000.

Sol: select NAME, SALES, JOBTITLE from employee, job where employee.jobid = job.jobid and sales > 1300000;

iii) To display names and corresponding job titles of those employees who have ‘SINGH’ (anywhere) in their names.

Sol: select NAME, JOBTITLE from employee, job where employee.jobid = job.jobid and NAME like '%SINGH%';

iv) Identify foreign key in the table EMPLOYEE.

Sol: desc employee;

v) Write SQL command to change the JOBID to 104 of the EMPLOYEE with ID as E4 in the table ‘EMPLOYEE’.

Sol: update employee set JOBID = 104 where EMPLOYEEID = "E4";