

# S20230010225\_Lab7

## DBMS LAB 7 Queries and solutions

Example queries:

```
4 • create procedure sar()
5   begin
6     select count(*) from borrower;
7   end aa
8   delimiter ;
9
10 • call sar();
```

Result Grid	Filter Rows:	Export:	Wrap Cell Content:
count(*)			
8			

Result 7 ×

Output

Action Output

#	Time	Action	Message
✓ 1	14:55:52	create procedure sar() begin select count(*) from borrower; end	0 row(s) affected
✓ 2	14:55:52	call sar()	1 row(s) returned

Drop procedure:

```
12 • drop procedure sar;
```

Output

☰ Action Output ▾

#	Time	Action	Message
✓ 1	14:55:52	create procedure sar() begin select count(*) from borrower; end	0 row(s) affected
✓ 2	14:55:52	call sar()	1 row(s) returned
✓ 3	14:56:57	call sar	1 row(s) returned
✓ 4	14:57:36	drop procedure sar	0 row(s) affected

Variables:

```

12 delimiter //
13 • create procedure sar()
14 begin
15     declare a, b int default 0;
16     set a=10;
17     select count(*) +a into b from borrower;
18     select b;
19 end//
20 delimiter ;
21 • call sar();

```

Result Grid	Filter Rows:	Export:	Wrap Cell Content:
b			
18			

Result 9 x

Output

Action Output

#	Time	Action	Message
13	15:04:48	create procedure sar() begin declare a, b int default 0; set a=10; select count(*) +a int...	0 row(s) affected
14	15:04:48	call sar()	1 row(s) returned

## IN Parameter

```

22 • drop procedure sar;
23
24 delimiter //
25 • create procedure sar(in name varchar(10))
26 begin
27     select * from customer where name=customername;
28 end//
29
30 delimiter ;
31 • call sar('johnson');

```

Result Grid	Filter Rows:	Export:	Wrap Cell Content:
customername	customerstreet	customercity	
Johnson	Alma	Palo Alto	

Result 10 x

Output

Action Output

#	Time	Action	Message	Duration
22	15:27:09	create procedure sar(in name varchar(10)) begin select * fr...	0 row(s) affected	0.000 se
23	15:27:09	call sar(johnson)	1 row(s) returned	0.016 se

## OUT parameter

```

33     delimiter //
34 •   create procedure temp(in balance int, out count int)
35     begin
36     select count(*) into count from account where account.balance>=balance;
37     end//
38
39     delimiter ;
40
41 •   call temp(1000, @a);
42 •   select @a;

```

Result Grid | Filter Rows: | Export: | Wrap Cell Content: |

@a
0

Result 11 x | Read C

Output

Action Output

#	Time	Action	Message	Duration / Fetch
31	15:34:27	call temp(1000, @a)	1 row(s) affected	0.000 sec
32	15:34:27	select @a LIMIT 0, 1000	1 row(s) returned	0.000 sec / 0.000 sec

## INOUT parameter

```

45     delimiter //
46 •   create procedure setcounter(inout counter int, in increment int)
47     begin
48     set counter = counter + increment ;
49     end//
50     delimiter ;
51 •   set @counter=1;
52 •   call setcounter(@counter,1);
53 •   call setcounter(@counter,10);
54 •   select @counter;

```

Result Grid | Filter Rows: | Export: | Wrap Cell Content: |

@counter
12

Result 12 x | Read C

Output

Action Output

#	Time	Action	Message	Duration / Fetch
36	15:39:54	set @counter=1	0 row(s) affected	0.000 sec

If statements:

```

58 begin
59 declare bal numeric(12,2) default 0;
60 select balance into bal from account where accountnumber=id; if bal > 700 then
61 set level='PLATINUM';
62 elseif bal<=700 and bal>300 then
63 set level='GOLD';
64 else
65 set level='SILVER';
66 end if;

```

Result Grid	Filter Rows:	Export:	Wrap Cell Content:
@var			
GOLD			

#	Time	Action	Message
44	15:44:14	create procedure customerlevel(in id varchar(10), out level varchar(20)) begin declare bal nu...	0 row(s) affected
45	15:44:14	call customerlevel('A-101', @var)	1 row(s) affected
46	15:44:14	select @var LIMIT 0, 1000	1 row(s) returned

Switch case statements:

```

72
73 delimiter //
74 • create procedure account_check(in id varchar(10),out status varchar(10))
75 begin
76 declare temp varchar(10) default 'Absent';
77 case id
78 when 'A-101' then
79 set status='Present';
80 select status;
81

```

Result Grid	Filter Rows:	Export:	Wrap Cell Content:
@var			
Absent			

#	Time	Action	Message
47	15:46:40	create procedure account_check(in id varchar(10),out status varchar(10)) begin declare tem...	0 row(s) affected
48	15:46:40	call account_check('A101', @var)	0 row(s) affected
49	15:46:40	select @var LIMIT 0, 1000	1 row(s) returned

IF case statements

```

102     delimiter //
103 •   create procedure customerlevel2( in id varchar(10),out level varchar(20))
104     begin
105     declare bal numeric(12,2) default 0;
106     select balance into bal from account where accountnumber = id;
107     case
108     when bal>700 then
109     set level 'PLATINUM';

```

Result Grid

Filter Rows:  Export: Wrap Cell Content:

@a
GOLD

Result 15 x

Output

Action Output

#	Time	Action	Message
✓ 54	15:48:35	create procedure customerlevel2( in id varchar(10),out level varchar(20)) begin declare ba...	0 row(s) affected
✓ 55	15:48:35	call customerlevel2('A-101',@a)	1 row(s) affected
✓ 56	15:48:35	select @a LIMIT 0, 1000	1 row(s) returned

## Loops:

```

136     end if;
137     end loop;
138
139     select str;
140     end //
141
142     delimiter ;
143
144 •   call looperdemo();

```

Result Grid

Filter Rows:  Export: Wrap Cell Content:

str
2,4,6,8,10,

Result 16 x

Output

Action Output

#	Time	Action	Message	Duration / Fe
✗ 57	15:53:26	create procedure looperdemo() begin declare x int default ...	Error Code: 1064. You have an error in your SQL syntax;...	0.000 sec
✓ 58	15:53:54	create procedure looperdemo() begin declare x int default ...	0 row(s) affected	0.016 sec
✓ 59	15:54:17	call looperdemo()	1 row(s) returned	0.000 sec / 1

## ACTUAL QUESTIONS:

### Q1

```
146 delimiter //
147 • create procedure q1()
148 • begin
149 • insert into loan values('L-100', 'Sricity', 1000);
150 • end//
151
152 delimiter ;
153 • call q1();
154 • select * from loan where branch_name='sricity';
```

Result Grid | Filter Rows:  | Export: | Wrap Cell Content:

	loan_number	branch_name	amount
▶	L-100	Sricity	1000

loan 19 x

Output

Action Output

#	Time	Action	Message	Duration / Fetch
✓ 9	16:02:13	create procedure q1() begin insert into loan values('L-10...	0 row(s) affected	0.015 sec
✓ 10	16:02:13	call q1()	1 row(s) affected	0.000 sec
✓ 11	16:02:13	select * from loan where branch_name='sricity' LIMIT 0, ...	1 row(s) returned	0.000 sec /

### Q2

```
156 • drop procedure q1;
```

Output

Action Output

#	Time	Action	Message	Duration / Fetch
▶ 10	16:02:13	call q1()	1 row(s) affected	0.000 sec
▶ 11	16:02:13	select * from loan where branch_name='sricity' LIMIT 0, ...	1 row(s) returned	0.000 sec / 0.000
▶ 12	16:02:59	drop procedure q1	0 row(s) affected	0.016 sec

### Q3

```

159 • create procedure q3(in a int, in b int, out sum int, out mul int)
160 • begin
161 •   set sum=a+b;
162 •   set mul=a*b;
163 • end//
164
165   delimiter ;
166 • call q3(5, 10, @sum, @mul);
167 • select @sum, @mul;

```

Result Grid		Filter Rows:	Export:	Wrap Cell Content:
	@sum	@mul		
15	15	50		

Result 20					
Output					
Action Output					
#	Time	Action	Message	Duration / Fe	
13	16:07:24	create procedure q3(in a int, in b int, out sum int, out mul...	0 row(s) affected	0.015 sec	
14	16:07:24	call q3(5, 10, @sum, @mul)	0 row(s) affected	0.000 sec	
15	16:07:24	select @sum, @mul LIMIT 0, 1000	1 row(s) returned	0.000 sec /	

## Q4

```

169   delimiter //
170 • create procedure q4(in number int)
171 • begin
172 •   select number*number;
173 • end//
174
175   delimiter ;
176 • call q4(10);

```

Result Grid		Filter Rows:	Export:	Wrap Cell Content:
	number*number			
16	100			

Result 21					
Output					
Action Output					
#	Time	Action	Message	Duration / Fe	
16	16:09:05	call q4(10)	Error Code: 1305. PROCEDURE s20230010225_lab7.q...	0.000 sec	
17	16:09:11	create procedure q4(in number int) begin select number*...	0 row(s) affected	0.016 sec	
18	16:09:11	call q4(10)	1 row(s) returned	0.000 sec /	

## Q5

```
177
178     delimiter //
179 •   create procedure q5(in str varchar(100))
180     begin
181     select char_length(str);
182     end//
183     delimiter ;
184
185 •   call q5('dbmslab7')
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

	char_length(str)
▶	8

Result 22 x

Output

Action Output

	#	Time	Action	Message	Duration
✓	18	16:09:11	call q4(10)	1 row(s) returned	0.000 se
✓	19	16:13:48	create procedure q5(in str varchar(100)) begin select ch...	0 row(s) affected	0.016 se
✓	20	16:13:48	call q5('dbmslab7')	1 row(s) returned	0.016 se

## Q6



```

188 • create procedure q6(in no int)
189   begin
190     declare factorial int default 1;
191     while no>0 do
192       set factorial = factorial * no;
193       set no=no-1;
194     end while;
195     select factorial;
196   end //
197   delimiter ;

```

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

factorial
120

Result 23 x

Output

Action Output

#	Time	Action	Message	Duration
✓ 21	16:20:31	create procedure q6(in no int) begin declare factorial int ...	0 row(s) affected	0.015 se
✓ 22	16:20:31	call q6(5)	1 row(s) returned	0.000 se

## Q7

```

delimiter //
create procedure q7(in old varchar(100), in new varchar(100))
begin
update customer set customername = new where customername=old;
end //

```

```

delimiter ;
call q7('adams', 'rahul');

```

on Output

Time	Action	Message	Duration / Fetch
16:31:38	create procedure q7(in old varchar(100), in new varchar...	0 row(s) affected	0.031 sec
16:31:38	call q7('adams', 'rahul')	1 row(s) affected	0.000 sec

## Q8

```

14 select case
15     when exists (select * from borrower b join customer2 c on b.customer_name = c.customername
16                 where c.customerid = id)
17     then 'Loan exists'
18     else 'No Loan'
19     end as loan_exists;
20 end//
21 delimiter ;
22
23 • call q8('192-83-7465');

```

Result Grid | Filter Rows: | Export: | Wrap Cell Content: [IA](#)

loan_exists
No Loan

## Q9

```

226 • create procedure q9(in id int, in name varchar(255), in street varchar(255), in city varchar(255),
227     in accnum int)
228     begin
229         declare exist int;
230         select count(*) into exist from customer2 where customerid = id;
231         if exist > 0 then
232             select customer_city from customer2 where customerid = id;
233         else
234             insert into customer2 values (id, name, street, city, accnum);
235             select 'New customer added';
236         end if;
237     end//
238     delimiter ;
239 • call q9(1, 'Shriansh', 'iiits', 'sriciyt', 101);

```

Result Grid | Filter Rows: | Export: | Wrap Cell Content: [IA](#)

New customer added
--------------------

▶ New customer added

## Q10

```

241     delimiter //
242 •   create procedure q10(in minbal decimal(10,2))
243     begin
244         select accountnumber from account where balance > minbal;
245     end//
246     delimiter ;
247 •   call q10(100)
248

```

Result Grid		Filter Rows:	Export:	Wrap Cell Content:
	accountnumber			
▶	A-101			
	A-102			
	A-201			
	A-215			
	A-217			

Result 26 x Read Only

Output

#	Time	Action	Message	Duration / Fetch
✓ 1	17:01:10	create procedure q10(in minbal decimal(10,2)) begin	sel... 0 row(s) affected	0.016 sec
✓ 2	17:01:10	call q10(100)	7 row(s) returned	0.000 sec / 0.000 sec

## SQL CODE:

```
-- create database S20230010225_LAB7
```

```
-- delimiter aa
```

```
-- create procedure sar()
```

```
-- begin
```

```
-- select count(*) from borrower;
```

```
-- end aa
```

```
-- delimiter ;
```

```
-- call sar;
```

```
-- delimiter //
```

```
-- create procedure sar()
```

```
-- begin

-- declare a, b int default 0;

-- set a=10;

-- select count(*) +a into b from borrower;

-- select b;

-- end//

-- delimiter ;

-- call sar();

-- drop procedure sar;


-- delimiter //

-- create procedure sar(in name varchar(10))

-- begin

-- select * from customer where name=customername;

-- end//


-- delimiter ;

-- call sar('johnson');


-- delimiter //

-- create procedure temp(in balance int, out count int)

-- begin

-- select count(*) into count from account where account.balance>=balance;

-- end//


-- delimiter ;

-- call temp(1000, @a);
```

```
-- select @a;
```

```
-- delimiter //
```

```
-- create procedure setcounter(inout counter int, in increment int)
```

```
-- begin
```

```
-- set counter = counter + increment ;
```

```
-- end//
```

```
-- delimiter ;
```

```
-- set @counter=1;
```

```
-- call setcounter(@counter,1);
```

```
-- call setcounter(@counter,10);
```

```
-- select @counter;
```

```
-- delimiter //
```

```
-- create procedure customerlevel( in id varchar(10), out level varchar(20))
```

```
-- begin
```

```
-- declare bal numeric(12,2) default 0;
```

```
-- select balance into bal from account where accountnumber=id; if bal > 700 then
```

```
-- set level='PLATINUM';
```

```
-- elseif bal<=700 and bal>300 then
```

```
-- set level='GOLD';
```

```
-- else
```

```
-- set level='SILVER';
```

```
-- end if;
```

```
-- end //
```

```
-- delimiter ;
```

```
-- call customerlevel('A-101', @var);

-- select @var;


-- delimiter //

-- create procedure account_check(in id varchar(10),out status varchar(10))

-- begin

-- declare temp varchar(10) default 'Absent';

-- case id

-- when 'A-101' then

-- set status='Present';

-- Select status;

-- when 'A-102' then

-- set status='Present';

-- when 'A-201' then

-- set status='Present';

-- when 'A-215' then

-- set status='Present';

-- when 'A-217' then

-- set status='Present';

-- when 'A-222' then

-- set status='Present';

-- when 'A-305' then

-- set status='Present';

-- else

-- Set status='Absent';

-- end case;

-- end //
```

```

-- delimiter ;

-- call account_check('A101', @var);

-- select @var;


-- delimiter //

-- create procedure customerlevel2( in id varchar(10),out level varchar(20))

-- begin

-- declare bal numeric(12,2) default 0;

-- select balance into bal from account where accountnumber = id;

-- case

-- when bal>700 then

-- set level='PLATINUM';

-- when bal<=700 and bal>300 then

-- set level='GOLD';

-- else

-- set level='SILVER';

-- end case;

-- end //

-- delimiter ;

-- call customerlevel2('A-101',@a);

-- select @a;


-- delimiter //


-- create procedure loopydemo()

-- begin

-- declare x int default 1;

-- declare str varchar(255) default "";

```

```
-- loop_label: loop
-- if x>10 then
-- leave loop_label;
-- end if;
-- set x= x+1;
-- if (x mod 2) then
-- iterate loop_label;
-- else
-- set str=concat(str, x, ',');
-- end if;
-- end loop;

-- select str;
-- end //

-- delimiter ;

-- call loopydemo();

-- delimiter //
-- create procedure q1()
-- begin
-- insert into loan values('L-100', 'Sricity', 1000);
-- end//

-- delimiter ;
-- call q1();
```



```
-- select * from loan where branch_name='srlicity';

-- drop procedure q1;

-- delimiter //
-- create procedure q3(in a int, in b int, out sum int, out mul int)
-- begin
-- set sum=a+b;
-- set mul=a*b;
-- end//

-- delimiter ;
-- call q3(5, 10, @sum, @mul);
-- select @sum, @mul;

-- delimiter //
-- create procedure q4(in number int)
-- begin
-- select number*number;
-- end//

-- delimiter ;
-- call q4(10);

-- delimiter //
-- create procedure q5(in str varchar(100))
-- begin
-- select char_length(str);
```

```
-- end//

-- delimiter ;

-- call q5('dbmslab7')

-- delimiter //

-- create procedure q6(in no int)
-- begin
-- declare factorial int default 1;
-- while no>0 do
-- set factorial = factorial * no;
-- set no=no-1;
-- end while;
-- select factorial;
-- end //

-- delimiter ;

-- call q6(5);

-- delimiter //

-- create procedure q7(in old varchar(100), in new varchar(100))
-- begin
-- update customer set customername = new where customername=old;
-- end //

-- delimiter ;

-- call q7('adams', 'rahul');

-- drop procedure q8;
```

```

-- delimiter //

-- create procedure q8(in id varchar(100))

-- begin

--   select case

--     when exists (select * from borrower b join customer2 c on b.customer_name =
c.customername

--       where c.customerid = id)

--     then 'Loan exists'

--     else 'No Loan'

--   end as loan_exists;

-- end//

-- delimiter ;

-- call q8('192-83-7465');

-- delimiter //

-- create procedure q9(in id int, in name varchar(255), in street varchar(255), in city
varchar(255),

-- in accnum int)

-- begin

-- declare exist int;

-- select count(*) into exist from customer2 where customerid = id;

-- if exist > 0 then

-- select customer_city from customer2 where customerid = id;

-- else

-- insert into customer2 values (id, name, street, city, accnum);

-- select 'New customer added';

-- end if;

```

```
-- end//  
  
-- delimiter ;  
  
-- call q9(1, 'Shriansh', 'iiits', 'sriciyt', 101);  
  
  
-- delimiter //  
  
-- create procedure q10(in minbal decimal(10,2))  
  
-- begin  
  
--   select accountnumber from account where balance > minbal;  
  
-- end//  
  
-- delimiter ;  
  
-- call q10(100)
```