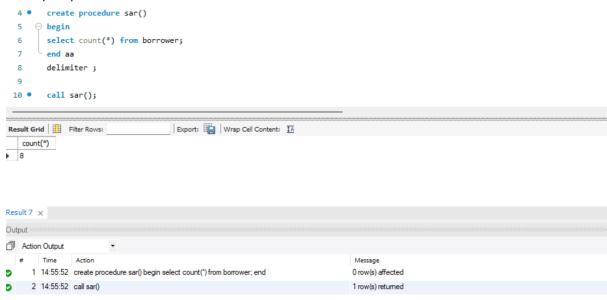
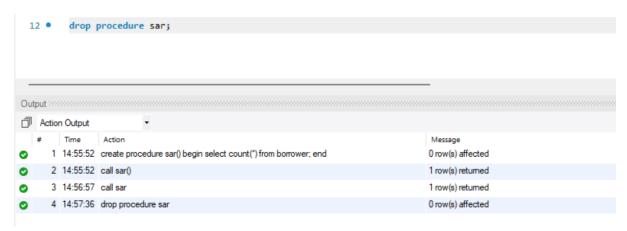
S20230010225_Lab7

DBMS LAB 7 Queries and solutions

Example queries:



Drop procedure:



Variables:

```
12
      delimiter //
 13 • create procedure sar()
 15
        declare a, b int default 0;
        set a=10;
 17
        select count(*) +a into b from borrower;
 18
        select b;
      end//
 19
      delimiter ;
 20
 21 • call sar();
Result Grid Filter Rows:
                                    Export: Wrap Cell Content: IA
  b
18
Result 9 ×
Output ::
Action Output
 # Time Action
                                                                             Message
3 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
       delimiter //
 24
 25 • create procedure sar(in name varchar(10))
       select * from customer where name=customername;
 27
 28
      end//
 29
 30 delimiter;
 31 • call sar('johnson');
Result Grid Filter Rows:
                                   Export: Wrap Cell Content: IA
   customername customerstreet customercity
Johnson
              Alma
                           Palo Alto
```



OUT parameter

```
delimiter //
33
34 •
     create procedure temp(in balance int, out count int)
35
36
      select count(*) into count from account where account.balance>=balance;
37
      end//
38
      delimiter;
39
40
41 • call temp(1000, @a);
42 •
     select @a;
                                  Export: Wrap Cell Content: IA
@a
0
esult 11 ×

    Read O
```

Message

1 row(s) affected

1 row(s) returned

Duration / Fetch

0.000 sec / 0.000 sec

0.000 sec

INOUT parameter

31 15:34:27 call temp(1000, @a)

32 15:34:27 select @a LIMIT 0, 1000

il Action Output

Time Action

```
delimiter //

de create procedure setcounter(inout counter int, in increment int)

begin

set counter = counter + increment;

end//

delimiter;

set @counter=1;

set @counter=1;

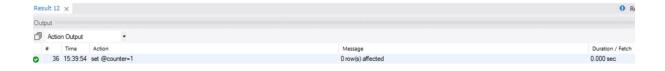
call setcounter(@counter,1);

acall setcounter(@counter,10);

set ecounter:

Result Grid 
Filter Rows:

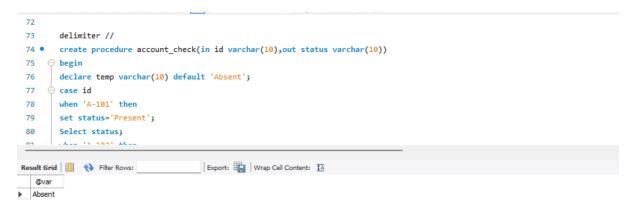
| Export: | Wrap Cell Content: | IA
```



If statements:

```
58 ♥ begin
59
       declare bal numeric(12,2) default 0;
60 \ominus select balance into bal from account where accountnumber=id; if bal > 700 then
       set level='PLATINUM';
61
       elseif bal<=700 and bal>300 then
62
63
       set level='GOLD';
64
       else
65
       set level='SILVER';
       end if;
Export: Wrap Cell Content: TA
 @var
 GOLD
esult 13 ×
utput :
Action Output
             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:





IF case statements

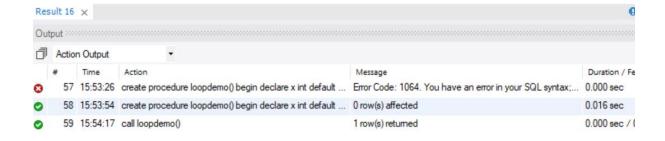
```
delimiter //
 102
 103 • create procedure customerlevel2( in id varchar(10),out level varchar(20))
 104 ⊖ begin
 105
          declare bal numeric(12,2) default 0;
          select balance into bal from account where accountnumber = id;
 106
 107
 108
          when bal>700 then
                                            Export: Wrap Cell Content: IA
Result Grid Filter Rows:
@a
• GOLD
Result 15 ×
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 loopdemo();
                                    Export: Wrap Cell Content: IA
Result Grid Filter Rows:
   str
2,4,6,8,10,
```



ACTUAL QUESTIONS:

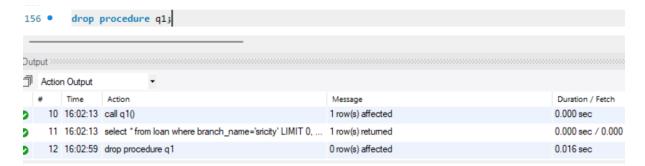
Q1

```
delimiter //
146
147 •
       create procedure q1()

→ begin

148
149
        insert into loan values('L-100', 'Sricity', 1000);
150
        end//
151
152
        delimiter;
153 •
       call q1();
       select * from loan where branch_name='sricity';
                                     Export: Wrap Cell Content: IA
loan_number branch_name amount
▶ L-100
             Sricity
                        1000
```





```
159 •
         create procedure q3(in a int, in b int, out sum int, out mul int)

    ⇒ begin

160
         set sum=a+b;
161
162
         set mul=a*b;
         end//
163
164
         delimiter;
165
166 •
         call q3(5, 10, @sum, @mul);
167 •
         select @sum, @mul;
Export: Wrap Cell Content: IA
                                                                                                                 @sum @mul
          50
15
Result 20 ×
Output :::::
Action Output
                                                             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
          delimiter //
169
          create procedure q4(in number int)

    ⇒ begin

171
          select number*number;
172
173
          end//
174
175
          delimiter;
          call q4(10);
Result Grid | | Filter Rows:
                                        Export: Wrap Cell Content: IA
    number*number
100
Result 21 ×
Output :::
Action Output
```

Message

1 row(s) returned

Error Code: 1305. PROCEDURE s20230010225_lab7.q... 0.000 sec

Duration / Fe

0.016 sec

0.000 sec /

Time

16 16:09:05 call q4(10)

18 16:09:11 call q4(10)

Action

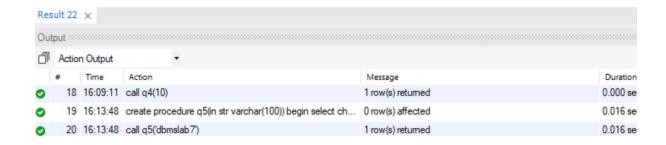
17 16:09:11 create procedure q4(in number int) begin select number*... 0 row(s) affected

Q5

```
1//
         delimiter //
178
         create procedure q5(in str varchar(100))
179 •
180

→ begin

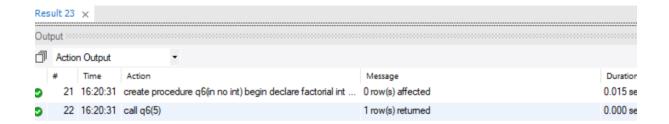
         select char_length(str);
181
         end//
182
183
         delimiter;
184
         call q5('dbmslab7')
185 •
Result Grid | Filter Rows:
                                      Export: Wrap Cell Content: TA
   char_length(str)
▶ 8
```



```
188 •
       create procedure q6(in no int)
189

    ⇒ begin

         declare factorial int default 1;
190
      ⇔ while no>0 do
191
         set factorial = factorial * no;
192
193
        set no=no-1;
194
        end while;
         select factorial;
195
       end //
196
         aunamaken .
Result Grid | Filter Rows:
                                     Export: Wrap Cell Content: IA
  factorial
120
```



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

```
\Diamond
14
          select case
15
               when exists (select * from borrower b join customer2 c on b.customer_name = c.customername
               where c.customerid = id)
16
17
               then 'Loan exists'
               else 'No Loan'
18
19
           end as loan_exists;
20
      end//
       delimiter;
21
22
23 •
      call q8('192-83-7465');
esult Grid Filter Rows:
                                Export: Wrap Cell Content: 1A
 loan_exists
No Loan
```

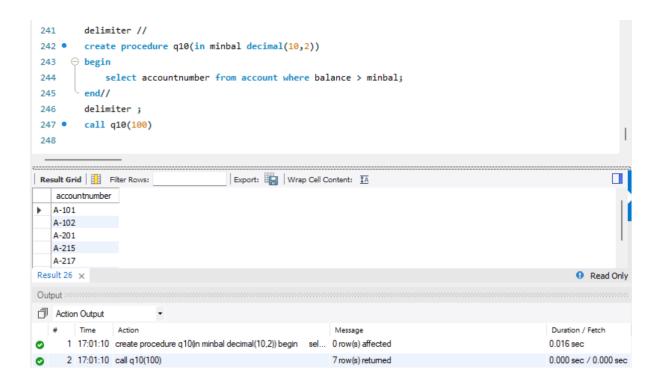
```
226 ● ⊖ create procedure q9(in id int, in name varchar(255), in street varchar(255), in city varchar(255),
      in accnum int)
227
228

⊖ begin

229
        declare exist int;
        select count(*) into exist from customer2 where customerid = id;

    if exist > 0 then

231
232
        select customer_city from customer2 where customerid = id;
233
234
        insert into customer2 values (id, name, street, city, accnum);
235
        select 'New customer added';
       end if;
236
       end//
237
238
        delimiter;
      call q9(1, 'Shriansh', 'iiits', 'sriciyt', 101);
239 •
Result Grid | Filter Rows:
                                    Export: Wrap Cell Content: IA
   New customer
   added
New customer added
```



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);
```

```
-- 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;
```

-- select @a;

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

```
-- select * from loan where branch_name='sricity';
-- 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)