```
/*
drop procedure if exists pro1;
delimiter $
create procedure pro1()
BEGIN
    select "Hello World" as "Mesage Box";
end $
delimiter ;


drop procedure if exists pro3;
delimiter $
create procedure pro3(x int,y int)
BEGIN
declare z1 int;
declare z2 int ;
set z1:=x+y;
set z2:=x-y;
select z1 as "z1";
select z2 as "z2";

end $
delimiter ;



drop procedure if exists pro5;
delimiter $
create PROCEDURE pro5(x int)
begin
        if x>10
        THEN
        select "Good" as "marks";
        else
        select "bad" as "marks";
        end if;
end $
delimiter ;


drop procedure if exists pro6;
delimiter $
create procedure pro6()
begin

        declare x int;
        set x:=0;
        lb11:LOOP
            set x:=x+1;
            select x;
            if x>10 THEN
            leave lb11;
            end if;
            end loop lb11;
            end $
            delimiter ;


drop procedure if exists pro7;
delimiter $
create procedure pro7()
begin
    declare x int;
    set x:=0;
    lb11:loop

    set x:=x+1;
```

```
    select x;
    if x>20 THEN
    leave lb11;
    end if;
    end loop lb11;
    end $
    delimiter ;



drop procedure if exists pro8;
delimiter $
create procedure pro8()
begin
        INSERT INTO proc1 VALUES(2,"MK","REDDY");
END $
DELIMITER ;







drop procedure if exists pro9;
delimiter $
create procedure pro9()
begin
     declare exit handler for 1062  select "Data already present"  as "error window";
     INSERT INTO proc1 VALUES(2,"MK","REDDY");
end $
    delimiter ;







drop procedure if exists pro9;
delimiter $
create procedure pro9()
begin
     declare exit handler for 1062  select "Data already present"  as "error window";
     INSERT INTO proc1 VALUES(3,"MK","REDDY");
     select "data successfully entered";
end $
    delimiter ;



drop procedure if exists pro10;
delimiter $
create procedure pro10(p1 int,p2 varchar(20),p3 varchar(30))
begin
    declare  exit handler for 1062 select "Data already present";
    if p3="Pune"
    THEN
    insert into proc2 values (p1,p2,p3);
    select "Data Succesfully inserted";
    else
    select "invalid location";
    end if;
    end $
    delimiter ;


drop procedure if exists pro11;
delimiter $
create procedure pro11(_deptno int )
```

```
begin
        declare flag bool;
            select true into flag from dept where deptno=_deptno;

            if flag then
            select * from dept where deptno=_deptno;
            else
            select "Record not found";
            end if;
            end $
            delimiter ;



drop procedure if exists pro12;
delimiter $
create procedure pro12(_deptno int,_empno int,ename varchar(30),_city varchar(30))
BEGIN

declare flag bool ;
        select true into flag from dept where _deptno=deptno;
        if flag then
        insert into emp values(_empno,_ename,_city,_deptno);
        select "record inserted successfully";
        else
        select "deptno not found into department table";
        end if;
end $
delimiter ;




drop procedure if exists pro9;
delimiter $
create procedure pro9(_tname varchar(20))
begin
    set @x:=concat("select* from ",_tname);
    prepare z from @x;
    execute z;
    end $
    delimiter ;




drop procedure if exists pro10;
delimiter $
create procedure pro10(_tname varchar(20))
begin
    set @x=concat("select * from ",_tname);
    prepare z from @x;
    execute z;
end $
delimiter ;




drop procedure if exists pro11;
delimiter $
create procedure pro11(_coname varchar(20),_tablename varchar(20))
begin
    set @x = concat('select ', _coname, ' from ', _tablename);
    prepare z from @x;
    execute z;
    end $
```

```
    delimiter ;

/*
DROP PROCEDURE IF EXISTS pro11;
DELIMITER $

CREATE PROCEDURE pro11(_coname VARCHAR(20), _tablename VARCHAR(20))
BEGIN
    SET @x = CONCAT('SELECT ', _coname, ' FROM ', _tablename);
    PREPARE stmt FROM @x;
    EXECUTE stmt;
    DEALLOCATE PREPARE stmt;
END $

DELIMITER ;




drop procedure if exists pro11;
delimiter $
create procedure pro11(_cname varchar(200),_tname varchar(200))
begin
    set @x=concat("select ", _cname ," from ", _tname );
    prepare z from @x;
    execute z;
end $
delimiter ;




__________FUNCTION_____________________



DROP FUNCTION IF EXISTS f1;
DELIMITER $

CREATE FUNCTION f1() RETURNS INT
DETERMINISTIC
BEGIN
    RETURN 10;
END $

DELIMITER ;


__________Procedure 1) asignment_________________


drop procedure if exists adduser;
delimiter $
create procedure adduser(username varchar(20),passward varchar(20),email varchar(20))
begin
 insert into login(username,passward,email) VALUES (username, passward, email);

end $
delimiter ;



```

```
_____________Procedure 2) assignment_____________
```

```
DROP PROCEDURE IF EXISTS checkuser;
DELIMITER $

CREATE PROCEDURE checkuser(_emailid VARCHAR(30))
BEGIN
    DECLARE p BOOLEAN;

    SELECT TRUE INTO p
    WHERE _emailid = (SELECT emailid FROM login);

    IF p THEN
        SELECT username, passward FROM login1 WHERE emailid = _emailid;
    ELSE
        INSERT INTO login1 (curr_date, curr_time, msg) VALUES (CURRENT_DATE(),
        CURRENT_TIME(), 'Default');
    END IF;
END $

DELIMITER ;
```

```
  drop procedure if exists pro11;
  delimiter $
  create procedure pro11(id int)
  begin
        set id:=5;
        select id;
    end $
    delimiter ;
```

```
drop procedure if exists  pro13;
delimiter $
create procedure pro13()
begin
    declare exit handler for 1050 select "Table already exists";
    create table c1(id int primary key auto_increment,name varchar(20));
end $
delimiter
```

```
drop procedure if exists pro14;
delimiter $
create procedure pro14()
begin
    declare exit handler for 1050 select "Table already present";
    create table c2(id int,ename varchar(20));
end $
delimiter ;
```

```
345
346
347
348  drop procedure if exists pro15;
349  delimiter $
350  create procedure pro15(_cname varchar(20),_tname varchar(20))
351  begin
352      set @x := concat("select  ",  _cname  ," from  ",_tname ) ;
353      prepare z from @x;
354      execute z;
355  end $
356  delimiter ;
357
358  drop procedure if exists pro16;
359  delimiter $
360  create procedure  pro16(tname varchar (20))
361  begin
362          set @x := concat("select * from  ",tname);
363          prepare z from @x;
364          execute z;
365  end $
366  delimiter ;
367
368
369  drop procedure if exists pro16;
370  delimiter $
371  create procedure pro16(p1 int,p2 int ,p3 int)
372  begin
373
374
375      set @x1 :=p1;
376      set @x2 :=p2;
377      set @x3 :=p3;
378      set @y :="insert into table1(id,eid,did) values(?,?,?)";
379  prepare z from @y;
380  execute z using @x1,@x2,@x3;
381
382  end $
383  delimiter ;
384
385
386
387
388  /*
389  DROP PROCEDURE IF EXISTS pro16;
390  DELIMITER $
391
392  CREATE PROCEDURE pro16(p1 INT, p2 INT, p3 INT)
393  BEGIN
394      SET @x1 := p1;
395      SET @x2 := p2;
396      SET @x3 := p3;
397      SET @y := "INSERT INTO table1 (id, eid, did) VALUES (?, ?, ?)";
398
399      PREPARE z FROM @y;
400      EXECUTE z USING @x1, @x2, @x3;
401
402      DEALLOCATE PREPARE z;
403  END $
404
405  DELIMITER ;
406
407
408
409
410  drop procedure if exists pro1;
411  delimiter $
412  create procedure pro1()
413  begin
```

```
    declare z int ;
    select max(sal) into z from emp;
    end $
    delimiter ;

drop function if exists f1;
delimiter $
create function f1() returns int
DETERMINISTIC
begin
call pro1()
return 10;
end $
delimiter ;

DROP PROCEDURE IF EXISTS pro1;
DELIMITER $

CREATE PROCEDURE pro1()
BEGIN
    DECLARE z INT;
    SELECT MAX(sal) INTO z FROM emp;
END $

DELIMITER ;

DROP FUNCTION IF EXISTS f1;
DELIMITER $

CREATE FUNCTION f1() RETURNS INT
DETERMINISTIC
BEGIN
    CALL pro1();
    RETURN 10;
END $

DELIMITER ;




drop procedure if exists pro1;
delimiter $
create procedure pro1()
begin
insert into d values (2,2,2);
end $
delimiter ;



drop procedure if exists pro1;
delimiter $
create procedure pro1()
begin
declare exit handler for 1062 select "Data already present " as "error window";
insert into d values (2,2,2);
end $
delimiter ;

drop procedure if exists pro1;
delimiter $
create procedure pro1()
begin

insert into d values(3,7,4);
end $
delimiter ;

```

```
drop procedure if exists pro1;
delimiter $
create procedure pro1()
begin

declare exit handler for 1062 select "Data already present " as "error window";
insert into d values(3,7,4);
end $
delimiter ;



drop procedure if exists pro1;
delimiter $
create procedure pro1(p1 int,p2 int,p3 int)
begin
declare exit handler for 1062 select "Data already present" as "Error Window";
insert into d values(p1,p2,p3);
end $
delimiter ;


drop procedure if exists pro1;
delimiter $
create procedure pro1()
begin

declare exit handler for 1050 select "table already prsent" ;
create table d(a int,b int,c int);


end $
delimiter ;


drop procedure if exists pro2;
delimiter $
create procedure pro2(_cname varchar(20),_tname varchar(20))
begin


 declare exit handler for 1054 select "invalid column";
 set @x:=CONCAT("SELECT ", _cname, " FROM ", _tname);
prepare z from @x;
execute z;
end $
delimiter ;

*/
```