Additional CURSOR programs

```
create table t1 (eno int,ename varchar(10),email varchar(20));
insert into t1 values (11,'raj','raj@gmail.com');
insert into t1 values (22,'jai','jai@yahoo.com')
insert into t1 values (33,'sam','sam@mvsrec.edu.in');
```

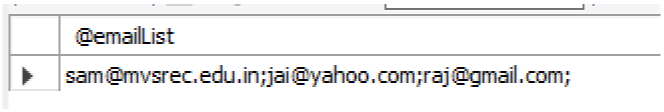
Procedure to concatenate email ids using Cursors:

```
delimiter $$

CREATE PROCEDURE createEmailList ( INOUT emailList  varchar(4000) )
BEGIN
    DECLARE finished INTEGER DEFAULT 0;
    DECLARE emailAddress varchar(100) DEFAULT "";
    -- declare cursor for employee email
    DEClARE curEmail CURSOR FOR SELECT email FROM t1;
    -- declare NOT FOUND handler
    DECLARE CONTINUE HANDLER FOR NOT FOUND SET finished = 1;
        OPEN curEmail;
                  getEmail: LOOP FETCH curEmail INTO emailAddress;
               IF finished = 1 THEN
                   LEAVE getEmail;
             END IF;
             -- build email list
           SET emailList = CONCAT(emailAddress,";",emailList);
             END LOOP getEmail;
         CLOSE curEmail;
END$$
delimiter ;
```

Execution steps

```
SET @emailList = "";
CALL createEmailList(@emailList);
SELECT @emailList;
```

Output:



```
@emailList
sam@mvsrec.edu.in;jai@yahoo.com;raj@gmail.com;
```

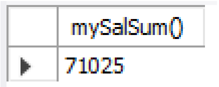
Function to find sum of salaries using Cursors:

```
DELIMITER $$

CREATE  FUNCTION mySalSum() RETURNS int(11)
BEGIN
  DECLARE record_not_found INTEGER DEFAULT 0;
  DECLARE lsal int;
  DECLARE s int DeFAULT 0;
  DECLARE c CURSOR FOR SELECT sal FROM emp;
  DECLARE CONTINUE HANDLER FOR NOT FOUND SET record_not_found = 1;
  OPEN c;
        edetails_loop: LOOP
              FETCH c INTO lsal;
              IF record_not_found THEN
                  LEAVE edetails_loop;
               END IF;
               set s=s+lsal;
            END LOOP edetails_loop;
 CLOSE c;
 RETURN s;
END$$

DELIMITER ;
```

Execution:

```
select mySalSum();
```

Output:



	mySalSum()
▶	71025