Assignment

Sept23/ DBT/126.1

Database Technologies

Diploma in Advance Computing

September 2023

**Procedure and Function**

|  |
| --- |
| 1. Write a procedure to accept a string and print all characters in separate lines.   Input: - Ram  Output: - R  a  m |
| drop procedure if exists p1;  delimiter $  create procedure p1(s varchar(20))  begin  declare s1 varchar(10);  declare x int;  declare y int;  set x = 1;  lbl:LOOP  select substr(s,x,1);  set x = x + 1;  if x > length(s) then  leave lbl;  end if;  end loop lbl;  end $  delimiter ; |
|  |
| 1. Write a procedure to accept a string and print every character separated by a comm sign.   Input: - SALEEL  Output: - S, A, L, E, E, L |
| drop procedure if exists pro2;  delimiter $  CREATE procedure pro2(s varchar(20))  begin  declare x int;  set x = 0;  set @z = '';  lbl:LOOP  set x = x + 1;  if x = 1 then  set @z := CONCAT(@z, substr(s,x,1));  ELSE  set @z := CONCAT(@z, ',', substr(s,x,1));  end if;  if x > length(s) - 1 THEN  leave lbl;  end if;  end LOOP lbl;  end $  delimiter ; |
|  |
| 1. Write a procedure to accept an alpha numeric string and separate number and characters of the string.   Input: - SAL1234EEL  Output: - SALEEL  1234 |
| drop procedure if exists pro2;  delimiter $  CREATE procedure pro2(s varchar(20))  begin  declare x int;  set x := 1;  set @num :="";  set @ch := "";  lbl:LOOP  if (substr(s, x, 1) >= '0' and substr(s, x, 1) <= '9') THEN  set @num := CONCAT(@num, '', SUBSTR(s,x,1));  ELSE  set @ch := CONCAT(@ch, '', SUBSTR(s,x,1));  end if;  set x := x + 1;  if x > length(s) THEN  leave lbl;  end if;  end LOOP lbl;  end $  delimiter ; |
|  |
| 1. Write a procedure to print all employee name and his job in following format.   Input: - ***KING PRESIDENT***  ***SCOTT ANALYST***  ***Output: - K(ING) is PRESIDENT***  **S(COTT) is ANALYST** |
| drop procedure if EXISTS pro2;  delimiter $  CREATE procedure pro2(name varchar(20),job varchar(20))  begin  declare z int;  set @x =name;  set @y =job;  set @z := concat(substr(@x, 1, 1), '(', substr(@x,2,length(@x)), ')', ' is ', @y);  end $  delimiter ; |
|  |
| 1. Write a procedure to print all upper and lower characters separately.   Input: - AbCdEfG  Output: - ACEG  bdf |
| drop procedure if exists pro2;  delimiter $  create procedure pro2(str varchar(20))  BEGIN  declare x int;  set @up = "";  set @low = "";  set x:= 0;  lbl:loop  if(ASCII(substr(str, x, 1)) >= 97 and ASCII(substr(str, x, 1)) <= 123) THEN  set @low := concat(@low, substr(str,x,1));  ELSE  set @up := concat(@up, substr(str,x,1));  end if;  set x = x + 1;  if x > length(str) THEN  Leave lbl;  end if;  end loop lbl;  end $  delimiter ; |
|  |
| 1. Write a procedure to find the number of vowels, digits and white spaces |
| drop PROCEDURE if exists pro1;  delimiter .  create PROCEDURE pro1(str1 varchar(50))  begin  declare x int;  set x=1;  set @digits=0;  set @vow=0;  set @spaces=0;  l:loop  if substr(str1,x,1)='a' or substr(str1,x,1)='e' or substr(str1,x,1)='i' or substr(str1,x,1)='o' or substr(str1,x,1)='u' THEN  set @vow=@vow+1;  end if;  if SUBSTR(str1,x,1)>='0' and SUBSTR(str1,x,1)<='9' THEN  set @digits=@digits+1;  end if;  if SUBSTR(str1,x,1)=' ' then set @spaces=@spaces+1;  end if;  set x=x+1;  if x>length(str1) then leave l;  end if;  end loop l;  end .  delimiter ; |
|  |
| 1. Write a procedure to remove all characters in a string except alphabets   Input: - saleel.bagde123@gmail.com  Output: - saleelbagdegmailcom |
| drop procedure if exists pro1;  delimiter .  create procedure pro1(str1 varchar(30))  begin  declare x int;  set x := 1;  set @input := '';  lbl:loop  if (ASCII(substr(str1, x , 1)) >= 97 and ASCII(substr(str1, x, 1)) <= 123) THEN  set @input = concat(@input, substr(str1, x, 1));  end if;  set x = x + 1;  if x>length(str1) then  leave lbl;  end if;  end loop lbl;  end .  delimiter ; |
|  |
| 1. Write a procedure to insert 10 rows in a table having following columns (using loop).   R (id int, message varchar(20)).  Output: -  id message  ---- -----------  1 i is odd  2 i is even  3 i is odd  4 i is even  5 i is odd  6 i is even  7 i is odd  8 i is even  9 i is odd  10 i is even |
| drop procedure if exists pro1;  delimiter @  create procedure pro1()  BEGIN  declare x int;  set x:= 1;  lbl:LOOP  if (mod(x,2) = 0) then  insert into r15 values ( concat(x, ' i is even'));  end if;  if (mod(x,2) = 1) then  insert into r15 values ( concat(x, ' i is odd'));  end if;  set x:= x + 1;  if x > 10 then leave lbl;  end if;  end loop lbl;  end @  delimiter ; |
|  |
| 1. Write a procedure to print five highest paid employees from the emp table using cursor. |
| drop procedure if EXISTS pro1;  delimiter .  create procedure pro1()  BEGIN  declare \_ename, \_job varchar(20);  declare sal int;  declare c1 cursor for select ename, job, sal from emp order by sal desc limit 5;  declare exit handler for 1329 select "EOF";  open c1;  lbl:LOOP  fetch from c1 into \_ename, \_job, sal;  select \_ename, \_job, sal;  end loop lbl;  close c1;  end .  delimiter ; |
|  |
| 1. Create the following table named (emp10, emp20, and emp30) which have the same structure of emp table.   Write a procedure to split employee records from emp table according to their department numbers and insert those records in the appropriate table using cursor. |
|  |
|  |
| 1. Write a procedure to display the department number and employee name in the following format.   Output: -  10 -> (AARAV, THOMAS, CLARK, KING, MILLER)  20 -> (SHARMIN, BANDISH, SMITH, JONES, SCOTT, FRED, ADAMS, FORD)  30 -> (GITA, ALLEN, WARD, MARTIN, BLAKE, TURNER, JAMES, HOFFMAN, GRASS)  40 –> (No employee work in department 40…)  50 -> (VRUSHALI, SANGITA, SUPRIYA) |
|  |
|  |
| 1. Write a procedure to accept customer number and display all his order. (Use customers and orders table) |
| drop procedure if exists p1;  delimiter $  create procedure p1(v int)  begin  DECLARE flag BOOL;  select true into flag from customers c where c.cnum=v;  if flag THEN  select c.cnum,cname,o.onum from customers c join orders o on c.cnum=o.cnum where o.cnum=v;  ELSE  select 'record not available';  end if;  end $  delimiter ; |
|  |
| 1. Write a procedure to convert numbers into word   Input: - 45234  Output: - Four Five Two Three Four |
|  |
|  |
| 1. Write a procedure to find the sum of digits.   Input: - 5675  Output: - Twenty Three |
|  |
|  |
| 1. Write a procedure to find how many “Sundays” are present between two given dates.   Input: - Date1 and Date2  Output: - 3 Sunday’s |
| drop procedure if exists pro1;  delimiter $  CREATE procedure pro1( d1 date,d2 date)  begin  declare x int;  set @cnt:=0;  set x=0;  set @dt:=0;  lb: LOOP  set @dt:=date(d1)+ interval x day;  if (dayofweek(@dt)=1) then  set @cnt:=@cnt+1;  end if;  set x=x+1;  if (datediff(@dt,d2) = 0) THEN  leave lb;  end if;  end loop lb;  end $  delimiter ; |
|  |
| 1. Writer a procedure which will accept date and weekday name from the user and print upcoming date on than weekday   Input: - (‘2023-04-26’, ‘Saturday’)  Output: - ‘2023-04-29’ |
|  |
|  |