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 string ;  delimiter $  create procedure string(string1 varchar(10))  begin  declare x int;  set x := 0;  lbl:LOOP  set x:=x+1;  select SUBSTR(string1,x,1);  if x>2 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 string2;  delimiter $  create procedure string2(string varchar(20))  BEGIN  declare str\_length int;  declare str\_char varchar(1);  declare i int;  set i:=1;  set str\_length:= length(string);  set @output:=" ";  lbl:loop  if i<=str\_length THEN  set str\_char:=substr(string,i,1);  set i:=i+1;  set @output := concat(@output,str\_char,",");  else  leave lbl;  end if;  end loop lbl;  select @output as output;  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 alphanum;  delimiter $  create procedure alphanum(alpha varchar(20))  BEGIN  declare x varchar(5);  declare i int;  declare str\_len int;  set @cap:="";  set @small:="";  set str\_len:=length(alpha);  set i:=0;  lbl:loop  set i:=i+1;  if i<=str\_len then  set x:=substr(alpha,i,1);  if ASCII(x)>=65 and ASCII(x)<= 90 then  set @cap:= concat(@cap,x);  else  set @small:=concat(@small,x);  end if;  else  leave lbl;  end if;  end loop lbl;  select @cap as "Letters";  select @small as "Numbers";  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 |
|  |
|  |
| 1. Write a procedure to print all upper and lower characters separately.   Input: - AbCdEfG  Output: - ACEG  bdf |
| drop procedure if exists alpha1;  delimiter $  create procedure alpha1(alpha varchar(20))  BEGIN  declare x varchar(5);  declare i int;  declare str\_len int;  set @cap:="";  set @small:="";  set str\_len:=length(alpha);  set i:=0;  lbl:loop  set i:=i+1;  if i<=str\_len then  set x:=substr(alpha,i,1);  if ASCII(x)>=65 and ASCII(x)<= 90 then  set @cap:= concat(@cap,x);  else  set @small:=concat(@small,x);  end if;  else  leave lbl;  end if;  end loop lbl;  select @cap as "Capital Letters";  select @small as "Small Letters";  end $  delimiter ; |
|  |
| 1. Write a procedure to find the number of vowels, digits and white spaces |
| drop procedure if exists cnt;  delimiter $  create procedure cnt(vow varchar(20))  BEGIN  declare i,str\_len int;  declare a,b,c,d varchar(5);  set @y:="";  set @p:="";  set @q:="";  set str\_len=LENGTH(vow);  set i=0,b=0,c=0,d=0;  lbl:LOOP  set i:=i+1;  if i<=str\_len THEN  set a:=substr(vow,i,1);  if a in ('a','e','i','o','u') then  set @y:=concat(@y,a);  set b:=b+1;  ELSEIF a in ('1','2','3','4','5','6','7','8','9','0') then  set @p:=concat(@p,a);  set c:=c+1;  ELSEIF ASCII(a)=32 then  set @q:=concat(@q,a);  set d:=d+1;  end if;  else  leave lbl;  end if;  end LOOP;  select b as 'Count\_Of\_Vowels';  select c as 'Count\_Of\_Numbers';  select d as 'Count\_Of\_Space';  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 alphanum;  delimiter $  create procedure alphanum(alpha varchar(50))  BEGIN  declare x varchar(5);  declare i int;  declare str\_len int;  set @cap:="";  set @small:="";  set str\_len:=length(alpha);  set i:=0;  lbl:loop  set i:=i+1;  if i<=str\_len then  set x:=substr(alpha,i,1);  if ASCII(x)>=97 and ASCII(x)<=122 then  set @cap:= concat(@cap,x);  end if;  else  leave lbl;  end if;  end loop lbl;  select @cap as "Output";  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 oddeven;  delimiter $  create procedure oddeven()  begin  declare i int;  set i:=0;  lbl:loop  set i:=i+1;  if i<=10 then  if mod(i,2)=1 THEN  insert into t (message) values("i is odd");  else  insert into t (message) values("i is even");  end if;  else  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 employee;  delimiter $  create procedure employee()  BEGIN  declare \_empno,\_ename,\_gender,\_job,\_mgr,\_sal varchar(20);  declare c1 cursor for select empno,ename,gender,job,mgr,sal from emp order by sal desc limit 5;  declare exit handler for 1329 select "No more record found ";  open c1;  lbl:loop  fetch c1 into \_empno,\_ename,\_gender,\_job,\_mgr,\_sal;  select \_empno,\_ename,\_gender,\_job,\_mgr,\_sal;  end loop;  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. |
| drop procedure if exists emp;  delimiter $  create procedure emp()  BEGIN  declare \_empno,\_deptno,\_mgr,\_sal varchar(50);  declare \_ename ,\_job,\_username varchar(50);  declare \_gender varchar(20);  declare c1 cursor for select empno,deptno,ename,gender,job,mgr,sal,'user name' from emp where deptno=10;  declare c2 cursor for select empno,deptno,ename,gender,job,mgr,sal,'user name' from emp where deptno=20;  declare c3 cursor for select empno,deptno,ename,gender,job,mgr,sal,'user name' from emp where deptno=30;  open c1;  open c2;  open c3;  lbl:loop  fetch c1 into \_empno,\_deptno,\_ename,\_gender,\_job,\_mgr,\_sal,\_username ;  insert into emp10 values(\_empno,\_deptno,\_ename,\_gender,\_job,\_mgr,\_sal,\_username );  select \_empno,\_deptno,\_ename,\_gender,\_job,\_mgr,\_sal,\_username from emp10 ;    fetch c2 into \_empno,\_deptno,\_ename,\_gender,\_job,\_mgr,\_sal,\_username ;  insert into emp20 values(\_empno,\_deptno,\_ename,\_gender,\_job,\_mgr,\_sal,\_username);  select \_empno,\_deptno,\_ename,\_gender,\_job,\_mgr,\_sal,\_username from emp20 ;      fetch c3 into \_empno,\_deptno,\_ename,\_gender,\_job,\_mgr,\_sal,\_username ;  insert into emp30 values(\_empno,\_deptno,\_ename,\_gender,\_job,\_mgr,\_sal,\_username);  select \_empno,\_deptno,\_ename,\_gender,\_job,\_mgr,\_sal,\_username from emp30 ;  end loop;  close c1;  close c2;  close c3;  end $  delimiter ; |
|  |
| 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 cust;  delimiter $  create procedure cust(cid int)  BEGIN  select o.\* from orders o join customers c on o.cnum=c.cnum where c.cnum=cid;  end $  delimiter ; |
|  |
| 1. Write a procedure to convert numbers into word   Input: - 45234  Output: - Four Five Two Three Four |
| drop procedure if exists number1;  delimiter $  create procedure number1(num int)  BEGIN  declare i int;  declare x int;  declare str\_len int;  set str\_len:= LENGTH(num);  set i:=0;  set @word:="";  lbl:LOOP  set i:=i+1;  if i<=str\_len then  set x:=substr(num,i,1);  IF x =0 THEN  set @word:=concat(@word,'Zero'," ");    ELSEIF x =1 THEN  set @word:=concat(@word,'One'," ");    ELSEIF x =2 THEN  set @word:=concat(@word,'Two'," ");    ELSEIF x =3 THEN  set @word:=concat(@word,'Three'," ");    ELSEIF x =4 THEN  set @word:=concat(@word,'Four'," ");    ELSEIF x =5 THEN  set @word:=concat(@word,'Five'," ");    ELSEIF x =6 THEN  set @word:=concat(@word,'Six'," ");    ELSEIF x =7 THEN  set @word:=concat(@word,'Seven'," ");    ELSEIF x =8 THEN  set @word:=concat(@word,'Eight'," ");    ELSEIF x =9 THEN  set @word:=concat(@word,'Nine'," ");  end if;  else  leave lbl;  end if;  end LOOP lbl;  select @word as "Output";  end $  delimiter ; |
|  |
| 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 |
|  |
|  |
| 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’ |
|  |
|  |