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(\_str varchar(10))**  **BEGIN**  **declare len1 int;**  **declare l int;**  **set l=1;**  **set len1=length(\_str);**  **1st:LOOP**  **select SUBSTR(\_str,l,1) as "string";**  **set l=l+1;**  **if l=len1+1 then**  **leave 1st;**  **end if;**  **end loop 1st;**  **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 p2();**  **delimiter $**  **CREATE PROCEDURE p2(\_str varchar(10))**  **BEGIN**  **declare len1 int;**  **declare l int;**  **set l=0;**  **set len1=length(\_str);**  **set @z="";**  **1st:LOOP**    **set l=l+1;**  **set @z=concat(@z,CONCAT(SUBSTR(\_str,l,1) , ","));**  **set @z=left(@z, len1+len1-1);**  **if l=len1+1 then**  **leave 1st;**  **end if;**    **end loop 1st;**  **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 p2();**  **delimiter $**  **CREATE PROCEDURE p2(\_str varchar(100))**  **BEGIN**  **declare len1 int;**  **declare l int;**  **set l=0;**  **set len1=length(\_str);**  **set @z="";**  **set @n="";**  **1st:LOOP**    **while l<=len1+1 do**  **set l=l+1;**  **if ascii(substr(\_str,l,1)) between 65 and 90 THEN**  **set @z=concat(@z,CONCAT(SUBSTR(\_str,l,1)));**  **ELSE**  **set @n=concat(@n,CONCAT(SUBSTR(\_str,l,1)));**    **end if;**  **end while;**  **leave 1st;**      **end loop 1st;**  **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 p2;  delimiter $  CREATE PROCEDURE p2(string1 VARCHAR(15))  BEGIN  DECLARE len\_str int;  DECLARE counter int;  DECLARE position varchar(20);  DECLARE ch varchar(2);  set @res:="";    set counter:=1;  set len\_str:= LENGTH(string1);    lbl1:LOOP  IF counter < len\_str+1 THEN  set ch:= SUBSTR(string1,counter,1);    IF counter =1 THEN  set @res:=CONCAT(ch,'(');  ELSEIF ch=" " THEN  set @res:=CONCAT(@res,") is ");  set position:=SUBSTR(string1,counter+1,len\_str);  leave lbl1;  ELSE  set @res:=CONCAT(@res,ch);  end IF;    set counter:=counter+1;  ELSE  leave lbl1;  end IF;  end LOOP lbl1;  select CONCAT(@res,position) as 'output';  end $  delimiter ; |
|  |
| 1. Write a procedure to print all upper and lower characters separately.   Input: - AbCdEfG  Output: - ACEG  bdf |
| **DROP PROCEDURE IF EXISTS demo3();**  **delimiter $**  **CREATE PROCEDURE demo3(\_str varchar(100))**  **BEGIN**  **declare len1 int;**  **declare l int;**  **set l=0;**  **set len1=length(\_str);**  **set @z="";**  **set @n="";**  **1st:LOOP**  **while l<=len1+1 do**  **set l=l+1;**  **if ascii(substr(\_str,l,1)) between 65 and 90 THEN**  **set @z=concat(@z,CONCAT(SUBSTR(\_str,l,1)));**  **end if;**  **IF ascii(substr(\_str,l,1)) between 97 and 122 THEN**  **set @n=concat(@n,CONCAT(SUBSTR(\_str,l,1)));**  **end if;**  **end while;**  **leave 1st;**  **end loop 1st;**  **end $**  **delimiter ;** |
|  |
| 1. Write a procedure to find the number of vowels, digits and white spaces |
| **DROP PROCEDURE IF EXISTS demo5();**  **delimiter $**  **CREATE PROCEDURE demo5(\_str varchar(100))**  **BEGIN**  **declare len1 int;**  **declare l,c1,c2,c3 int;**  **set l=0;**  **set len1=length(\_str);**  **set c1=0;**  **set c2=0;**  **set c3=0;**  **set @v="";**  **1st:LOOP**  **while l<=len1+1 do**  **set l=l+1;**  **set@v=(substr(\_str,l,1));**    **if (@v="a" or @v="e" or @v="i" or @v="o" or @v="u" or @v="A" or @v="E" or @v="I" or @v="O" or @v="U") THEN**  **set c1=c1+1;**  **end if;**    **if @v=" " THEN**  **set c2=c2+1;**  **end if;**    **if (@v between 1 and 9) THEN**  **set c3=c3+1;**  **end if;**    **end while;**  **leave 1st;**  **end loop 1st;**  **select c1 as " vowels", c2 as " whitespaces ",c3 as "digit";**  **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 demo4();**  **delimiter $**  **CREATE PROCEDURE demo4(\_str varchar(100))**  **BEGIN**  **declare len1 int;**  **declare l int;**  **set l=0;**  **set len1=length(\_str);**  **set @z="";**  **1st:LOOP**  **while l<=len1+1 do**  **set l=l+1;**  **if (ascii(substr(\_str,l,1)) between 65 and 90) or (ascii(substr(\_str,l,1)) between 97 and 122 )THEN**  **set @z=concat(@z,CONCAT(SUBSTR(\_str,l,1)));**  **end if;**  **end while;**  **leave 1st;**  **end loop 1st;**  **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**  **create table login5(\_id int , message varchar(20));**  **end $**  **delimiter ;**    **drop procedure if exists pro2;**  **delimiter $**  **create procedure pro2()**  **begin**  **declare x int;**  **set x=0;**  **lst :LOOP**  **while x<10 do**  **set x:=x+1;**  **if (x%2=1) then**  **insert into login5 values (x,"i is odd ");**  **else**  **INSERT into login5 values (x,"i is even ");**  **end if;**  **end while;**  **leave lst;**  **end loop lst;**    **select \* from login5;**  **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 \_sal int;**  **declare \_ename varchar(20);**  **declare c2 CURSOR for select ename,sal from emp order by sal desc limit 5;**  **declare exit handler for 1329 select "invalid";**  **open c2;**  **lbl:LOOP**  **fetch c2 into \_ename ,\_sal;**  **select \_ename,\_sal ;**  **end loop lbl;**  **close c2;**  **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. |
| **create table emp10 like emp;**  **create table emp20 like emp;**  **create table emp30 like emp;**  **drop procedure if exists pro1;**  **delimiter $**  **create procedure pro1()**  **begin**  **declare \_empno,\_sal,\_deptno,\_mgr ,\_comm,\_bonusid int;**  **declare \_ename,\_job,\_gender,\_username,\_pwd,\_phone varchar(20);**  **declare \_hiredate date;**  **declare \_isactive bool;**  **declare c2 CURSOR for select \* from emp ;**  **declare exit handler for 1329 select "invalid";**  **open c2;**  **lbl:LOOP**  **if DEPTNO=10 then**  **fetch c2 into \_empno,\_ename,\_gender,\_job,\_mgr,\_hiredate,\_sal,\_comm,\_deptno,\_bonusid,\_username,\_pwd,\_phone,\_isactive;**  **insert into emp10 values(\_empno,\_ename,\_gender,\_job,\_mgr,\_hiredate,\_sal,\_comm,\_deptno,\_bonusid,\_username,\_pwd,\_phone,\_isactive);**  **END IF;**  **if DEPTNO=20 then**  **insert into emp20 values(\_empno,\_ename,\_gender,\_job,\_mgr,\_hiredate,\_sal,\_comm,\_deptno,\_bonusid,\_username,\_pwd,\_phone,\_isactive);**  **END IF;**  **if DEPTNO=30 then**  **insert into emp30 values(\_empno,\_ename,\_gender,\_job,\_mgr,\_hiredate,\_sal,\_comm,\_deptno,\_bonusid,\_username,\_pwd,\_phone,\_isactive);**  **end if;**  **end LOOP lbl;**  **close c2;**  **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) |
| **drop procedure if exists format;**  **delimiter $**  **create procedure format()**  **BEGIN**  **select deptno,group\_concat(ename) from emp group by deptno;**  **end $**  **delimiter ;** |
|  |
| 1. Write a procedure to accept customer number and display all his order. (Use customers and orders table) |
| **drop procedure if exists CDEATILS;**  **delimiter $**  **create procedure CDETAILS(\_CNO int)**  **BEGIN**  **select c.CNUM,ONUM,Amt from customers c join orders o where c.CNUM=o.CNUM and c.CNUM=\_CNO;**  **END$**  **delimiter ;** |
|  |
| 1. Write a procedure to convert numbers into word   Input: - 45234  Output: - Four Five Two Three Four |
| **drop procedure if exists digitTOW;**  **delimiter $**  **create procedure digitTOW(\_str varchar(60))**  **BEGIN**  **declare len1 int;**  **declare i int;**  **set len1=length(\_str);**  **set @z="";**  **set i=0;**  **while i<=len1 do**  **set i=i+1;**  **set @c=ascii(substr(\_str,i,1));**  **if @c=48 then**  **set @z=concat(@z,"zero ");**  **end if;**  **if @c=49 then**  **set @z=concat(@z,"one ");**  **end if;**  **if @c=50 then**  **set @z=concat(@z,"two ");**  **end if;**  **if @c=51 then**  **set @z=concat(@z,"three ");**  **end if;**  **if @c=52 then**  **set @z=concat(@z,"four ");**  **end if;**  **if @c=53 then**  **set @z=concat(@z,"five ");**  **end if;**  **if @c=54 then**  **set @z=concat(@z,"six ");**  **end if;**  **if @c=55 then**  **set @z=concat(@z,"seven ");**  **end if;**  **if @c=56 then**  **set @z=concat(@z,"eight ");**  **end if;**  **if @c=57 then**  **set @z=concat(@z,"nine ");**  **end if;**  **END WHILE;**  **select @z as 'ANSWER';**  **end $**  **delimiter ;** |
|  |
| 1. Write a procedure to find the sum of digits.   Input: - 5675  Output: - Twenty Three |
| **drop PROCEDURE IF EXISTS p14;**  **delimiter $**  **CREATE PROCEDURE p14(num bigint)**  **BEGIN**  **DECLARE len\_str int;**  **DECLARE n int;**  **DECLARE res int;**  **set len\_str:= LENGTH(num);**    **set @counter:=1;**  **set res:=0;**    **lbl1:LOOP**  **IF @counter < len\_str+1 THEN**  **set n:= SUBSTR(num,@counter,1);**  **set res:= res + n;**    **set @counter:=@counter+1;**  **ELSE**  **leave lbl1;**  **end IF;**    **end LOOP lbl1;**  **call pro13(res);**  **end $**  **delimiter ;** |
|  |
| 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 p15;**  **delimiter $**  **create procedure p15(date1 date,date2 date)**  **begin**  **declare count int;**  **set count := 0;**  **lbl1:loop**  **if date1<date2 then**  **if date\_format(date1,'%W') = "Sunday" then**  **set count := count + 1;**  **set date1 := date1 + interval 1 day;**  **else**  **set date1 := date1 + interval 1 day;**  **end if;**  **else**  **leave lbl1;**  **end if;**  **end loop lbl1;**  **select count;**  **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’ |
| **drop procedure if exists p16;**  **delimiter $**  **create procedure p16(date1 date, weekday1 varchar(10))**  **begin**  **lbl1:loop**  **if date\_format(date1,'%W')=weekday1 then**  **select date1;**  **leave lbl1;**  **else**  **set date1 := date1+interval 1 day;**  **end if;**  **end loop lbl1;**  **end $**  **delimiter ;** |
|  |