Assignment

**PRN NO - 230945920032\_230945920091**

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 proa1;**  **delimiter $**  **create procedure proa1(str varchar(20))**  **BEGIN**    **declare j int;**  **set j := 1;**    **loop1: LOOP**    **if j<=length(str) THEN**    **select substr(str,j,1);**  **set j :=j+1;**  **else**  **leave loop1;**  **end if;**  **end LOOP loop1;**  **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 proa2;**  **delimiter $**  **create procedure proa2(str varchar(20))**  **BEGIN**    **declare j int;**  **declare s varchar(20);**  **declare r varchar(200);**  **set j := 1;**    **set r:= ' ';**    **loop1: LOOP**    **if j<=length(str) THEN**    **set s := substr(str,j,1);**  **set j :=j+1;**  **set r := CONCAT(r,s,',');**    **else**  **leave loop1;**  **end if;**  **end LOOP loop1;**  **select r;**  **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 proa3;**  **delimiter $**  **create procedure proa3(str varchar(20))**  **BEGIN**    **declare j int;**  **declare sub varchar(10);**  **declare str1 varchar(30);**  **declare str2 varchar(30);**  **set str1 :=' ';**  **set str2 :=' ';**    **set j := 1;**    **loop1: LOOP**    **if j<=length(str) THEN**    **set sub:= substr(str,j,1);**  **set j :=j+1;**    **if (ascii(sub)>= 97 and ascii(sub)<= 123) THEN**    **set str1 = CONCAT(str1,sub,',');**  **else**    **set str2 = CONCAT(str2,sub,',');**    **end if;**  **else**  **leave loop1;**  **end if;**  **end LOOP loop1;**    **select str1;**  **select str2;**    **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 proa3;**  **delimiter $**  **create procedure proa3(str varchar(20))**  **BEGIN**    **declare j int;**  **declare sub varchar(10);**  **declare str1 varchar(30);**  **declare str2 varchar(30);**  **set str1 :='';**  **set str2 :=' ';**    **set j := 1;**    **loop1: LOOP**    **if j<=length(str) THEN**    **set sub:= substr(str,j,1);**      **if j=2 THEN**    **set str1 := CONCAT(str1,'(',sub);**    **else IF sub= str2 THEN**  **set str1 := CONCAT(str1,')',' is ');**  **ELSE**    **set str1 := CONCAT(str1,sub);**  **end if;**  **end if;**  **set j :=j+1;**  **else**  **leave loop1;**  **end if;**  **end LOOP loop1;**    **select str1;**    **end $**  **delimiter ;** |
|  |
| 1. Write a procedure to print all upper and lower characters separately.   Input: - AbCdEfG  Output: - ACEG  bdf |
| **drop procedure if EXISTS proa5;**  **delimiter $**  **create procedure proa5(str varchar(20))**  **BEGIN**    **declare j int;**  **declare sub varchar(10);**  **declare str1 varchar(30);**  **declare str2 varchar(30);**  **set str1 :=' ';**  **set str2 :=' ';**    **set j := 1;**    **loop1: LOOP**    **if j<=length(str) THEN**    **set sub:= substr(str,j,1);**  **set j :=j+1;**    **if (ascii(sub)>= 65 and ascii(sub)<= 90) THEN**    **set str1 = CONCAT(str1,sub,'');**  **else**    **set str2 = CONCAT(str2,sub,'');**    **end if;**  **else**  **leave loop1;**  **end if;**  **end LOOP loop1;**    **select str1 as 'Upper Characters';**  **select str2 as 'Lower Characters';**    **end $**  **delimiter ;** |
|  |
| 1. Write a procedure to find the number of vowels, digits and white spaces |
| **drop procedure if EXISTS proa6;**  **delimiter $**  **create procedure proa6(str varchar(1000))**  **BEGIN**    **declare j int;**  **declare sub varchar(50);**  **declare v,d,s int;**  **set v=0;**  **set d=0;**  **set s=0;**      **set j := 1;**    **loop1: LOOP**    **if j<=length(str) THEN**    **set sub:= substr(str,j,1);**      **if (sub = 'a' or sub = 'e' or sub = 'i' or sub = 'o' or sub = 'u' or sub = 'A' or sub = 'E' or sub = 'I' or sub = 'O' or sub = 'U' )THEN**  **set v := v+1;**    **else IF (ascii(sub)>= 48 and ascii(sub)<=57) THEN**  **set d := d+1;**    **ELSE if (ascii(sub)= 32) THEN**  **set s := s+1;**    **end if;**  **end if;**  **end if;**  **set j :=j+1;**  **else**  **leave loop1;**  **end if;**  **end LOOP loop1;**  **select v;**  **select d;**  **select s;**    **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 proa7;**  **delimiter $**  **create procedure proa7(In X varchar(50))**  **BEGIN**  **declare email varchar(30);**  **declare i int;**  **declare sub varchar(50);**  **declare str1 varchar(30);**  **set email :=X;**  **set i:=1;**  **set str1 := ' ';**  **lbl:LOOP**  **if i<= length (email) then**  **set sub := substr(email,i,1);**  **if (ascii(sub) between 97 and 123)then**  **set str1 := CONCAT(str1,sub);**  **end if;**  **set i=i+1;**  **else**  **leave lbl;**  **end if;**  **end loop lbl;**  **select str1;**    **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 proa9;**  **delimiter $**  **create procedure proa9(X int)**  **begin**  **declare i int;**  **declare len1 int;**  **declare message varchar(50);**  **declare even varchar(40);**  **declare odd varchar(40);**    **set i:= 1;**  **set message := ' ';**  **set odd := 'odd';**  **set even := 'even';**  **set len1 := X;**    **lbl2: LOOP**    **if i>len1 then**  **leave lbl2;**  **else**  **if i % 2 = 0 THEN**    **insert into proa9 values(i, even);**    **ELSE**    **insert into proa9 values(i, odd);**    **end if;**    **end if;**  **set i := i+1;**    **END LOOP lbl2;**  **end $**  **delimiter ;** |
|  |
| 1. Write a procedure to print five highest paid employees from the emp table using cursor. |
| **drop procedure if exists proa11;**  **delimiter $**  **create procedure proa11()**  **BEGIN**  **DECLARE \_EMPID,\_ESAL INT;**  **DECLARE \_ENAME VARCHAR(20);**    **DECLARE c1 cursor for select empno, ename , sal from emp order by sal desc limit 5;**  **DECLARE EXIT handler for 1329 select "EOF";**  **open c1;**    **l1:LOOP**    **FETCH c1 into \_EMPID,\_ENAME,\_ESAL;**  **SELECT \_EMPID,\_ENAME,\_ESAL;**    **end loop l1;**  **close c1;**  **end $** |
|  |
| 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 proc1;**  **delimiter $**  **create procedure proc1()**  **begin**  **DECLARE \_EMPID,\_ESAL,\_dno INT;**  **DECLARE \_ENAME,\_job,\_gender VARCHAR(20);**    **DECLARE c2 cursor for select empno, ename, sal, job, gender, deptno from emp where deptno in(10,20,30);**    **DECLARE EXIT handler for 1329 select "EOF";**  **open c2;**    **l1:LOOP**    **fetch c2 into \_EMPID,\_ENAME,\_ESAL,\_job,\_gender, \_dno ;**    **if \_dno=10 then**  **insert into emp10 (empno,sal,deptno,ename,job,gender) values (\_EMPID,\_ESAL,\_dno,\_ENAME,\_job,\_gender);**  **else if \_dno=20 THEN**  **insert into emp20 (empno,sal,deptno,ename,job,gender) values (\_EMPID,\_ESAL,\_dno,\_ENAME,\_job,\_gender);**  **else if \_dno=30 THEN**  **insert into emp30 (empno,sal,deptno,ename,job,gender) values (\_EMPID,\_ESAL,\_dno,\_ENAME,\_job,\_gender);**    **end if;**  **end if;**  **end if;**  **end loop l1;**    **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 proc1;**  **delimiter $**  **create procedure proc1()**  **begin**  **DECLARE \_dno,i INT;**  **DECLARE \_ENAME VARCHAR(2000);**  **declare str1,str2,str3,str4,str5 varchar(1000);**    **DECLARE c1 cursor for select distinct(deptno), group\_concat(ename) from emp group by deptno;**      **set @str1 := '';**  **set @str2 := '';**  **set @str3 := '';**  **set @str4 := '';**  **set @str5 := '';**  **OPEN c1;**    **loop1: LOOP**    **FETCH c1 into \_dno,\_ENAME;**      **IF \_dno = 10 THEN**  **set @str1 := CONCAT(@str1,',',\_ename);**  **select @str1;**    **else if \_dno = 20 THEN**  **set @str2 := CONCAT(@str2,',',\_ename);**  **select @str2;**    **else if \_dno = 30 THEN**  **set @str3 := CONCAT(@str3,',',\_ename);**  **select @str3;**    **else if \_dno = 40 THEN**  **set @str4 := CONCAT(@str4,',',\_ename);**  **select @str4;**    **else if \_dno = 50 THEN**  **set @str5 := CONCAT(@str5,',',\_ename);**  **select @str5;**  **end if;**  **end if;**  **end if;**  **end if;**  **end if;**  **end loop loop1;**  **CLOSE c1;**  **end $**  **delimiter ;** |
|  |
| 1. Write a procedure to accept customer number and display all his order. (Use customers and orders table) |
| **drop procedure if exists proa12;**  **delimiter $**  **create procedure proa12(\_xcnum int)**  **begin**    **DECLARE \_onum,\_cnum int;**  **DECLARE \_AMT float;**  **DECLARE \_ODATE date;**    **DECLARE c1 CURSOR for select ONUM, AMT, ODATE , CNUM from orders where CNUM=\_xcnum;**    **open c1;**    **l1:LOOP**    **FETCH c1 into \_onum, \_AMT, \_ODATE , \_cnum;**    **select \_onum, \_AMT, \_ODATE , \_cnum;**    **end loop l1;**  **close c1;**  **end $**  **delimiter ;** |
|  |
| 1. Write a procedure to convert numbers into word   Input: - 45234  Output: - Four Five Two Three Four |
| **drop procedure if exists pro13;**  **delimiter $**  **create procedure pro13(num int)**  **begin**  **declare inWords varchar(50);**  **declare x int;**  **set inWords := " ";**  **loop1:loop**  **if num > 0 then**  **set x := num % 10;**  **set num := num div 10;**  **if x = 1 then**  **set inWords := concat("one ",inWords);**  **elseif x = 2 then**  **set inWords := concat("two ",inWords);**  **elseif x = 3 then**  **set inWords := concat("three ",inWords);**  **elseif x = 4 then**  **set inWords := concat("four ",inWords);**  **elseif x = 5 then**  **set inWords := concat("five ",inWords);**  **elseif x = 6 then**  **set inWords := concat("six ",inWords);**  **elseif x = 7 then**  **set inWords := concat("seven ",inWords);**  **elseif x = 8 then**  **set inWords := concat("eight ",inWords);**  **elseif x = 9 then**  **set inWords := concat("nine ",inWords);**  **elseif x = 0 then**  **set inWords := concat(" ",inWords);**  **end if;**  **else**  **leave loop1;**  **end if;**  **end loop loop1;**  **select inWords;**  **end $**  **delimiter ;** |
|  |
| 1. Write a procedure to find the sum of digits.   Input: - 5675  Output: - Twenty Three |
| **drop function if exists firstNumber;**  **delimiter $**  **create function firstNumber(x bigint) returns varchar(20)**  **deterministic**  **begin**  **if x = 1 then**  **return "one";**  **elseif x = 2 then**  **return "two";**  **elseif x = 3 then**  **return "three";**  **elseif x = 4 then**  **return "four";**  **elseif x = 5 then**  **return "five";**  **elseif x = 6 then**  **return "six";**  **elseif x = 7 then**  **return "seven";**  **elseif x = 8 then**  **return "eight";**  **elseif x = 9 then**  **return "nine";**  **elseif x = 0 then**  **return " ";**  **end if;**  **end $**  **delimiter ;**  **drop procedure if exists pro14;**  **delimiter $**  **create procedure pro14(number1 int)**  **begin**  **declare inWords varchar(50);**  **declare num int;**  **declare temp int ;**  **declare tensPlace int;**  **set inWords := " ";**  **set num := 0;**  **loop1:loop**  **if number1 > 0 then**  **set temp := number1 % 10;**  **set number1 := number1 div 10;**  **set num := num + temp;**  **else**  **leave loop1;**  **end if;**  **end loop;**  **set tensPlace := num div 10;**  **if tensPlace = 2 then**  **set inWords := concat(inWords,"twenty");**  **elseif tensPlace = 3 then**  **set inWords := concat(inWords,"thirty");**  **elseif tensPlace = 4 then**  **set inWords := concat(inWords,"forty");**  **elseif tensPlace = 5 then**  **set inWords := concat(inWords,"fifty");**  **elseif tensPlace = 6 then**  **set inWords := concat(inWords,"sixty");**  **elseif tensPlace = 7 then**  **set inWords := concat(inWords,"seventy");**  **elseif tensPlace = 8 then**  **set inWords := concat(inWords,"eighty");**  **elseif tensPlace = 9 then**  **set inWords := concat(inWords,"ninty");**  **else**  **set inWords := concat(inWords," ");**  **end if ;**  **if tensPlace = 1 then**  **if num = 10 then**  **set inWords := concat(inWords,"ten");**  **elseif num = 11 then**  **set inWords := concat(inWords,"eleven");**  **elseif num = 12 then**  **set inWords := concat(inWords,"twelve");**  **elseif num = 13 then**  **set inWords := concat(inWords,"thirteen");**  **elseif num = 14 then**  **set inWords := concat(inWords,"forteen");**  **elseif num = 15 then**  **set inWords := concat(inWords,"fifteen");**  **elseif num = 16 then**  **set inWords := concat(inWords,"sixteen");**  **elseif num = 17 then**  **set inWords := concat(inWords,"seventeen");**  **elseif num = 18 then**  **set inWords := concat(inWords,"eighteen");**  **elseif num = 19 then**  **set inWords := concat(inWords,"nineteen");**  **end if;**  **end if;**  **if tensplace != 1 then**  **set inWords := concat(inWords," ",firstNumber(num % 10));**  **end if;**  **select inWords;**  **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 pro15;**  **delimiter $**  **create procedure pro15(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 pro16;**  **delimiter $**  **create procedure pro16(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 ;** |
|  |