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 pro2;  delimiter $  CREATE PROCEDURE pro2(name VARCHAR(20))  BEGIN  DECLARE a INTEGER;  set a=0;  lbl:LOOP  set a:=a+1;  select substr(name, a ,1);  if a>length(name)-1 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 pro1;  delimiter $  create procedure pro1(str varchar(20))  BEGIN  declare x int;  set @y:="";  set x:=1;  lb1:LOOP  if x=1 then  SET @y:= CONCAT(@y,'',substr(str,x,1)) ;  else  SET @y:= CONCAT(@y,',',substr(str,x,1)) ;  end if;  set x=x+1;  if x>length(str) THEN  leave lb1 ;  end if;  end loop lb1;  end $  select @y;  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 pro1;  delimiter $  create procedure pro1(in name varchar(20))  BEGIN  declare x int;  set x := 1;  set @ch = "";  set @num = "";    lbl:Loop  if(substr(name,x,1) >='0' and (substr(name,x,1))<='9') THEN  set@num:=CONCAT(@num,'',substr(name,x,1));  ELSE  SET@ch:=CONCAT(@ch,'',substr(name,x,1));    end if;  set x := x+1;    if x > length(name) then leave lbl;  end if;  end loop lbl;  select @ch;  select @num;  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 pro11;  delimiter $  create procedure pro11()  BEGIN  select concat(substr(ename,1,1),'(',substr(ename,2),')', " is " , job) from emp;  end $  delimiter ; |
|  |
| 1. Write a procedure to print all upper and lower characters separately.   Input: - AbCdEfG  Output: - ACEG  bdf |
| drop procedure if exists pro11;  delimiter $  create procedure pro11(str1 varchar(50))  BEGIN  declare x int;  set x := 1;  set @lch="";  set @sch="";  lbl : loop  if ascii(substr(str1,x,1)) >= ascii('A') and ascii(substr(str1,x,1))<= ascii('Z') then  set @lch = concat(@lch, substr(str1,x,1));  ELSE  set @sch = concat(@Sch, 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 find the number of vowels, digits and white spaces |
| drop procedure if exists pro11;  delimiter $  create procedure pro11(name varchar(20))  BEGIN  declare x int;  set x := 1;  set @vowels = 0;  set @numbers = 0;  set @spaces = 0;    lbl :LOOP  if(substr(name,x,1) = 'a' or substr(name,x,1) = 'e' or substr(name,x,1) = 'i'  or substr(name,x,1) = 'o' or substr(name,x,1) = 'u' ) THEN  set @vowels = @vowels + 1;  end if;    if(substr(name,x,1) = " ") THEN  set @spaces = @spaces +1;  end if;    if(substr(name,x,1) >= '1' and substr(name,x,1) <= '9') THEN  set @numbers = @numbers + 1;  end IF;  set x := x+1;    if x>length(name) then leave lbl;  end if;    end loop lbl;  end $  delimiter ; |
| drop procedure if exists pro11;  delimiter $  create procedure pro11(name varchar(20))  BEGIN  declare x int;  set x := 1;  set @vowels = 0;  set @numbers = 0;  set @spaces = 0;    lbl :LOOP  if(substr(name,x,1) = 'a' or substr(name,x,1) = 'e' or substr(name,x,1) = 'i'  or substr(name,x,1) = 'o' or substr(name,x,1) = 'u' ) THEN  set @vowels = @vowels + 1;      elseif(substr(name,x,1) = " ") THEN  set @spaces = @spaces +1;      elseif(substr(name,x,1) >= '1' and substr(name,x,1) <= '9') THEN  set @numbers = @numbers + 1;      elseif x>length(name) then leave lbl;  end if;  set x := x+1;    end loop lbl;  select @vowels;  select @numbers;  select @spaces;  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 pro11;  delimiter $  create procedure pro11(name varchar(20))  BEGIN  declare x int;  set x = 1;  set @name = "";  lbl : LOOP  set @chars := substring(name,x,1);  if (ascii(@chars) >= 65 and ascii(@chars) <= 90) or (ascii(@chars) >= 97 and ascii(@chars) <= 122) then  set @name := concat(@name,@chars);  end if;  set x = x+1;  if x>length(name) then leave lbl;  end if;  end loop lbl;  select @name;  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 |
|  |
|  |
| 1. Write a procedure to print five highest paid employees from the emp table using cursor. |
|  |
|  |
| 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) |
|  |
|  |
| 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 |
|  |
|  |
| 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’ |
|  |
|  |