**DBMS SKILL 10**

**PRE-LAB**

1. **Analyze the code and tell your observation?**  
     **DECLARE  
    a number(3) := 100;  
    BEGIN  
    IF (a = 50 ) THEN  
    dbms\_output.put\_line('Value of a is 10' );  
    ELSEIF ( a = 75 ) THEN  
    dbms\_output.put\_line('Value of a is 20' );  
    ELSE  
    dbms\_output.put\_line('None of the values is matching');  
    END IF;  
    dbms\_output.put\_line('Exact value of a is: '|| a );  
    END;**

Ans) The Output is

None of the values is matching

Exact value of a is 100

**2. What will be the output of the following code?**

**DECLARE  
 lines dbms\_output.chararr;  
 num\_lines number;  
BEGIN  
 Dbms\_output.enable;  
 dbms\_output.put\_line('Hello!');  
 dbms\_output.put\_line('Hope you are doing well!');  
 num\_lines := 2;  
 dbms\_output.get\_lines(lines, num\_lines);  
 FOR i IN 1..num\_lines LOOP  
 dbms\_output.put\_line(lines(i));  
 END LOOP;  
END;**

**Ans)** Hello Reader

Hope you have enjoyed doing well

2

**3. Consider the following code :−**

**DECLARE  
 -- Global variables  
 num number := 95;  
BEGIN  
 dbms\_output.put\_line('num: ' || num1);  
 DECLARE  
 -- Local variables  
 num number := 195;  
 BEGIN  
 dbms\_output.put\_line('num: ' || num1);  
 END;  
END;**

**What will happen when the code is executed?**

Ans) Not executed , because syntax error.

1. **What would be printed when the following code is executed?**

**DECLARE  
 x NUMBER;  
 BEGIN  
 x := 5;  
 x := 10;  
 dbms\_output.put\_line(-x);  
 dbms\_output.put\_line(+x);  
 x := -10;  
 dbms\_output.put\_line(-x);  
 dbms\_output.put\_line(+x);  
 END;**

Ans) -10

10

10

-10

1. **What will be printed by the following PL/SQL block?**

**DECLARE  
 a number;  
 b number;  
 c number;  
PROCEDURE findMin(x IN number, y IN number, z OUT number) IS  
BEGIN  
IF x < y THEN  
 z:= x;  
 ELSE  
 z:= y;  
END IF;  
END;  
BEGIN  
 a:= 2;  
 b:= 5;  
 findMin(a, b, c);  
 dbms\_output.put\_line(c);  
END;**

Ans) -5

-10

-25

1. **What will be printed by the following PL/SQL block?**

**DECLARE  
 a number;  
PROCEDURE squareNum(x IN OUT number) IS  
BEGIN  
 x := x \* x;  
END;  
BEGIN  
 a:= 5;  
 squareNum(a);  
 dbms\_output.put\_line(a);  
END;**

**Ans) -5**

**-10**

**-25**

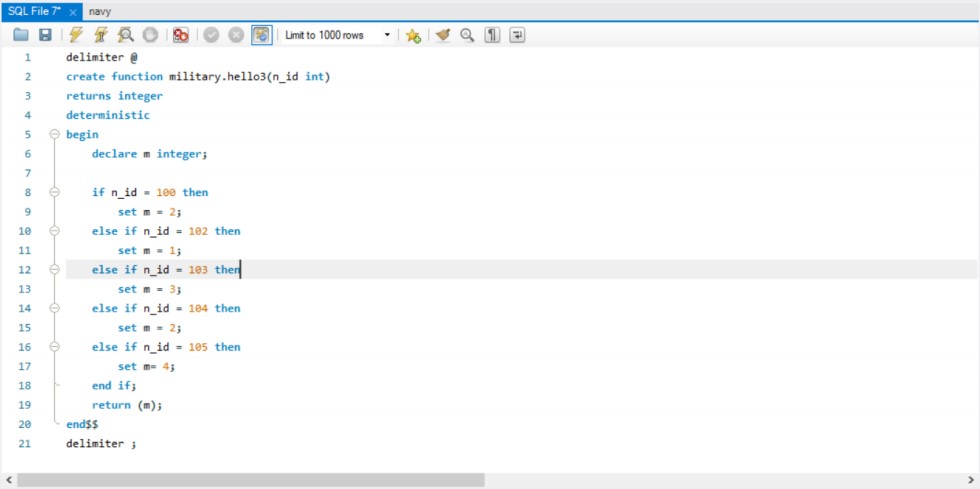
1. **When is the pre-defined exception “CASE\_NOT\_FOUND” raised?**

None of the choices in the when clauses of a case statement is selected , and there is no ELSE clause.

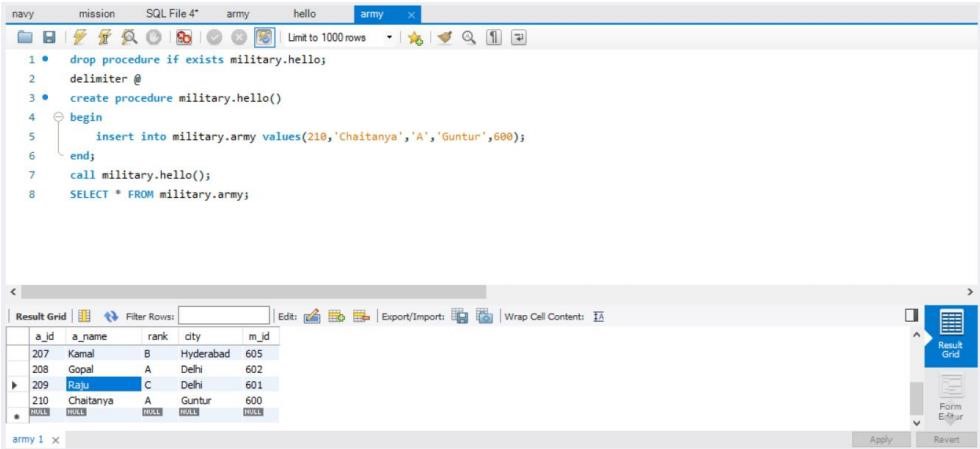
**INLAB**

Implement PL/SQL Queries on Case Study 9 (MILITARY DATABASE)

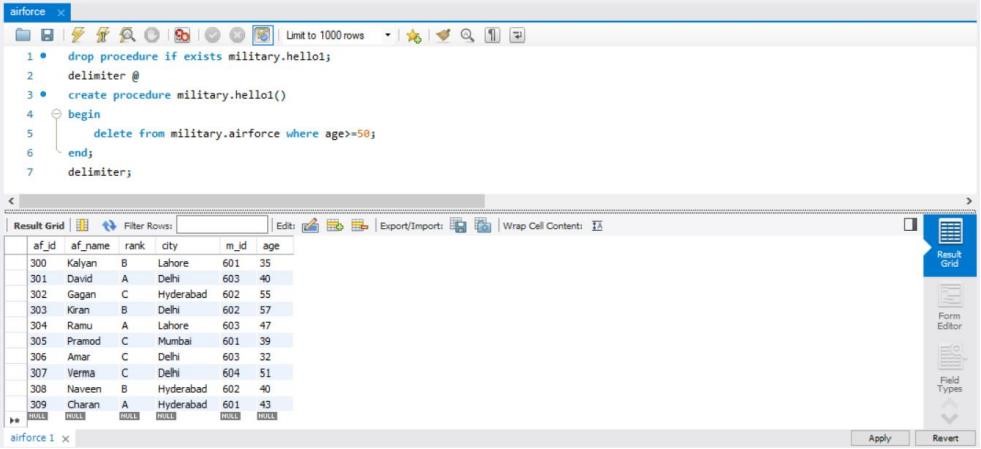
1. Create a function that takes solider ID FROM NAVY and return the number of mission done by the soldier in the past.

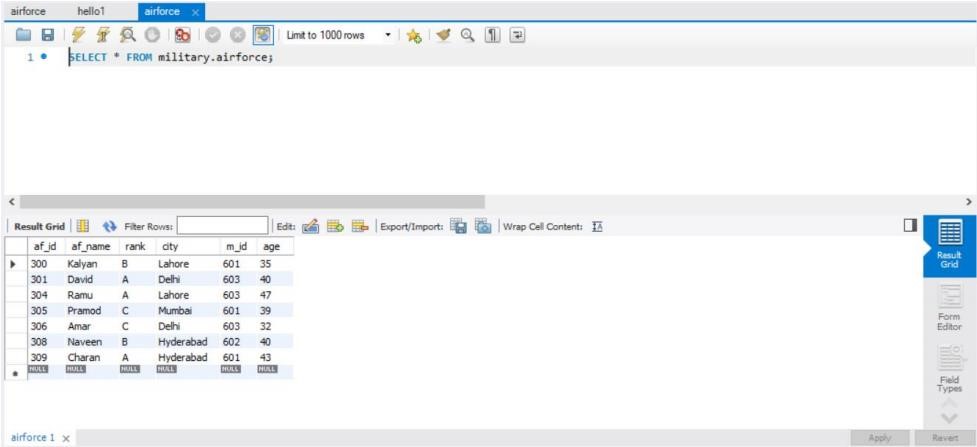


1. Create a procedure to insert record into army table



1. Create a procedure to delete airforce soldier of age above 50





**POSTLAB**

1. Create a user-defined exception by the name of exp\_check. Select the ename and hiredate of all employees into a cursor. Your program should calculate the experience of all the employees in years, and insert the ename and experience of each employee into temp table. If any employee has experience less than 2 years, the program should be aborted with a suitable message. Raise the user-defined exception exp\_check to achieve this. Display the results on the screen using dbms\_output.put\_line.

**Ans)**

**Declare exp\_check exception;**

**Cursor c1 IS**

**Select empname;**

**Joined**

**From emp;**

**Begin**

**For I in c1 loop**

**if (trunc(months\_between(sysdate, in joined)/12)<2) then raise exp\_check;**

**Else**

**Insert into temp2 Values ( I empname , trunc( months\_between(sysdate, I joined)/12));**

**End if;**

**End loop;**

**Excpetion**

**when exp\_check then**

**dbms\_output.put\_line(‘experience is less than 2 years not allowed’);**

**When others then**

**dbms\_output.put\_line(‘Unidentified error occurred’);**

**End;**

**Output: experience is less than 2 years is not allowed**

1. Create a table EMPLOYEE with the following columns:-

Employee No. Varchar24

Employee Name Varchar30

Designation Varchar10

Category Character 1

Basic Salary Number 4

Category may be ?J?, ?S?, or ?W? for Jr. officers, Sr. officers or Worker category.

Formulae:-

DA = 35% of Basic Salary correct up to paise.

HRA = 15% of Basic Salary subject to a maximum of Rs. 250/1000/30000 for categories W/J/S

respectively.

Gross = Basic Salary +DA +HRA

Output the Employee Number and the Gross for each employee in a separate table.

CREATE TABLE EMPLOYEE ( "Employee No " Varchar2 (4), "Employee Name" Varchar2 (30), Designation Varchar2 (10), Category Character (1), "Basic Salary " Number (4) )

--Insert dummy data:-

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| --Empl | EmployeeName | DESIGNATION | C | Basic Salary |
| --1000 | rakesh | Sr.off | s | 3000 |
| --1001 | peeyoosh | Sr.off | s | 4000 |
| --1002 | malik | Jr.off | s | 5000 |

CREATE TABLE employee\_gross

CREATE TABLE gross ( "Employee No " Varchar2 (4), "Gross Salary " Number (4) )

**Ans)**

**create or replace procedure gross as**

**cursor c1 is select \* from employee;**

**emp\_record employee%rowtype;**

**da number(20,2);**

**hra number(20,2);**

**gross number(20,2);**

**begin**

**delete from employee\_gross;**

**for emp\_record in c1**

**loop**

**da:=emp\_record."Basic Salary "\*35/100;**

**hra:=emp\_record."Basic Salary "\*15/100;**

**if emp\_record.Category='j' and hra>250 then**

**hra:=250;**

**elsif emp\_record.Category='s' and hra>1000 then**

**hra:=1000;**

**elsif emp\_record.Category='w' and hra>30000 then**

**hra:=30000;**

**else**

**hra:=0;**

**end if;**

**gross:=emp\_record."Basic Salary "+da+hra;**

**insert into employee\_gross values(emp\_record."Employee No ",gross);**

**end loop;**

**end;**

1. Write a program to read in a number and print it out digit by digit, as a series of words. For example, the number 523 would be printed as "five two three". Use decode function within a for loop. Display the results on the screen using dbms\_output.put\_line.

**Ans)**

**Declare**

**Num varchar(10):=’&number’;**

**I varchar(1);**

**C int := length(num);**

**Result varchar(10);**

**Begin**

**dbms\_output.put\_line(‘Entered no.Is’);**

**Loop i:=substr(num,I,1);**

**Num := substr(num,2);**

**Select decode(i,1,’one’,2,’two’,3,’three’,4,’four’,5,’five’,6,’six’,7,‘seven’,8,’eight’,9,’nine’,0,’zero’) into result 0 from dual;**

**dbms\_output.put\_line(Result); Exit when c=1;**

**c:=c-1;**

**end loop; end;**