Name:Mennatallah Abdelhmid

SQL and PL/SQL Labs

**PL/SQL (Day1):**

|  |  |
| --- | --- |
| *1* | * 1. *Create a function called* ***f\_dept*** *to take a* ***department number*** *as a parameter and return a* ***department name.*** |
|  | *create or replace function f\_dept(department\_number number)*  *return varchar2*  *is*  *dept\_name departments.department\_name%type;*  *BEGIN*  *select department\_name*  *into dept\_name*  *from departments*  *where department\_id=department\_number;*  *RETURN dept\_name;*  *end;* |
|  | * 1. *Compile the code; create a host variable* ***G\_name*** *and invoke the function with department* ***ID 20****.*   *Print the host variable to view the result.* |
|  | ***SQL> variable x varchar2(20);***  ***SQL> exec :x :=f\_dept(20);***  ***PL/SQL procedure successfully completed.***  ***SQL> print :x;***  ***X***  ***--------------------------------------------------------------------------------***  ***Marketing*** |
| *2* | *Create a procedure* ***update\_comm*** *to accept* ***employee number*** *as a parameter.*  *if employee salary is less than 1000 set the commission amount of the employee to 10%, if employee salary is between 1000 and 2000 set the commission amount of employee to 15%, if employee salary is greater than 2000 set the commission amount of employee to 20%* |
|  | ***create or replace procedure update\_comm (employee\_number number)***  ***is***  ***sal employees.salary%type;***  ***begin***  ***select salary***  ***into sal***  ***from employees***  ***where employee\_number=employee\_id;***  ***if sal<1000 then***  ***update employees***  ***set commission\_pct=10/100***  ***where employee\_id=employee\_number;***  ***elsif (sal>1000 and sal<2000) then***  ***update employees***  ***set commission\_pct=15/100***  ***where employee\_id=employee\_number;***  ***elsif sal>2000 then***  ***update employees***  ***set commission\_pct=20/100***  ***where employee\_id=employee\_number;***  ***end if;***  ***end;*** |
| *3* | *Create a procedure* ***print\_location*** *that declares a record with the same structure as table locations, in the executable section, select the data of the country\_id = 'mx' and* ***print the street address, city, and the location\_id.*** |
|  | *create or replace procedure print\_location(cont\_id varchar2)*  *is*  *result locations%rowtype;*  *begin*  *select \**  *into result*  *from locations*  *where lower(country\_id)=cont\_id;*  *DBMS\_OUTPUT.PUT\_LINE('location: '||result.location\_id);*  *DBMS\_OUTPUT.PUT\_LINE('street address: '||result.street\_address);*  *DBMS\_OUTPUT.PUT\_LINE('city: '||result.city);*  *end;*  */*  *SQL> exec print\_location('mx');*  *location: 3200*  *street address: Mariano Escobedo 9991*  *city: Mexico City* |
| *4* | *Create a procedure* ***print\_timetable*** *to display the following output*  *1\*1=1*  *1\*2=2*  *…….*  *……..*  *10\*10=100* |
|  | ***create or replace procedure print\_timetable***  ***is***  ***x number;***  ***begin***  ***for i in 1..10 loop***  ***for j in 1..10 loop***  ***x:=j\*i;***  ***DBMS\_OUTPUT.PUT\_LINE(i || '\*' || j || '=' || x);***  ***end loop;***  ***end loop;***  ***end;***  ***/***  ***exec print\_timetable***  ***1\*1=1***  ***1\*2=2***  ***1\*3=3***  ***1\*4=4***  ***1\*5=5***  ***1\*6=6***  ***1\*7=7***  ***1\*8=8***  ***1\*9=9***  ***1\*10=10***  ***2\*1=2***  ***2\*2=4***  ***2\*3=6***  ***2\*4=8***  ***2\*5=10***  ***2\*6=12***  ***2\*7=14***  ***2\*8=16***  ***2\*9=18***  ***2\*10=20***  ***3\*1=3***  ***3\*2=6***  ***3\*3=9***  ***3\*4=12***  ***3\*5=15***  ***3\*6=18***  ***3\*7=21***  ***3\*8=24***  ***3\*9=27***  ***3\*10=30***  ***4\*1=4***  ***4\*2=8***  ***4\*3=12***  ***4\*4=16***  ***4\*5=20***  ***4\*6=24***  ***4\*7=28***  ***4\*8=32***  ***4\*9=36***  ***4\*10=40***  ***5\*1=5***  ***5\*2=10***  ***5\*3=15***  ***5\*4=20***  ***5\*5=25***  ***5\*6=30***  ***5\*7=35***  ***5\*8=40***  ***5\*9=45***  ***5\*10=50***  ***6\*1=6***  ***6\*2=12***  ***6\*3=18***  ***6\*4=24***  ***6\*5=30***  ***6\*6=36***  ***6\*7=42***  ***6\*8=48***  ***6\*9=54***  ***6\*10=60***  ***7\*1=7***  ***7\*2=14***  ***7\*3=21***  ***7\*4=28***  ***7\*5=35***  ***7\*6=42***  ***7\*7=49***  ***7\*8=56***  ***7\*9=63***  ***7\*10=70***  ***8\*1=8***  ***8\*2=16***  ***8\*3=24***  ***8\*4=32***  ***8\*5=40***  ***8\*6=48***  ***8\*7=56***  ***8\*8=64***  ***8\*9=72***  ***8\*10=80***  ***9\*1=9***  ***9\*2=18***  ***9\*3=27***  ***9\*4=36***  ***9\*5=45***  ***9\*6=54***  ***9\*7=63***  ***9\*8=72***  ***9\*9=81***  ***9\*10=90***  ***10\*1=10***  ***10\*2=20***  ***10\*3=30***  ***10\*4=40***  ***10\*5=50***  ***10\*6=60***  ***10\*7=70***  ***10\*8=80***  ***10\*9=90***  ***10\*10=100*** |
| *5* | 1. *Create and invoke the function* ***ANNUAL\_COMP****, passing in values for monthly salary and commission. Either or both values passed can be NULL, but the function should still return an annual salary, which is not NULL. The annual salary is defined by the basic formula:*   *(Sal\*12) + (commission\_pct\*salary\*12)* |
|  | *create or replace function ANNUAL\_COMP (sal employees.salary%type , commission employees.commission\_pct%type)*  *return number*  *is*  *result number;*  *begin*  *result :=nvl((Sal\*12),0) + (nvl(commission,0)\*nvl(sal,0)\*12) ;*  *return result;*  *end;*  */*  *SQL> exec :y:=ANNUAL\_COMP(100,0);*  *PL/SQL procedure successfully completed.*  *SQL> print y;*  *Y*  *--------------------------------------------------------------------------------*  *1200*  *SQL>* |
|  | 1. *Use the function in a SELECT statement against the EMPLOYEES table for department 20.* |
|  | *select annual\_comp(salary,commission\_pct)*  *from employees*  *where department\_id=20*  *SQL> /*  *ANNUAL\_COMP(SALARY,COMMISSION\_PCT)*  *----------------------------------*  *156000*  *72000* |