Homework Week #5
PL/SQL Virtual Training

1. What happens when Oracle encounters a runtime problem while executing a PL/SQL block?

2. What do you need to add to your PL/SQL block to address these problems?

3. List three advantages of handling exceptions within a PL/SQL block.

4. Run this PL/SQL code and then answer the questions that follow.

```
DECLARE
  v_jobid    employees.job_id%TYPE;
BEGIN
  SELECT job_id
    INTO v_jobid
    FROM employees
    WHERE department_id = 80;
END;
```

   A. What happens when you run the block?

   B. In your oswn words, explain what you can do to fix this problem.

   C. Modify the code to fix the problem. Use a TOO_MANY_ROWS exception handler.

   D. Run your modified code. What happens this time?

5. What are the three types of exceptions that can be handled in a PL/SQL block?

6. What is the difference in how each of these three types of exception is handled in the PL/SQL block?

7. Enter and run the following PL/SQL block. Look at the output and answer the following questions:

```
DECLARE
  v_number      NUMBER(6,2) := 100;
  v_region_id   wf_world_regions.region_id%TYPE;
  v_region_name wf_world_regions.region_name%TYPE;
BEGIN
  SELECT region_id, region_name INTO v_region_id, v_region_name
    FROM wf_world_regions
    WHERE region_id = 1;
  DBMS_OUTPUT.PUT_LINE('Region: ' || v_region_id ||
                       ' is: ' || v_region_name);
  v_number := v_number / 0;
END;
```

   A. What error message is displayed and why ?

   B. Modify the block to handle this exception and re-run your code. Now what happens and why?

   C. Modify the block again to change the WHERE clause to region_id = 29. Re-run the block. Now what happens and why?

8. Enter and run the following PL/SQL code. What output is displayed, and why? Save your code.

```
DECLARE
  v_last_name    employees.last_name%TYPE;
BEGIN
  SELECT last_name INTO v_last_name
    FROM employees
    WHERE employee_id = 100;
    /* This employee’s last name is King */
  DECLARE
    v_last_name    employees.last_name%TYPE;
  BEGIN
    SELECT last_name INTO v_last_name
      FROM employees
       WHERE employee_id = 107;
      /* This employee’s last name is Lorentz */
    DBMS_OUTPUT.PUT_LINE(v_last_name);
  END;
  DBMS_OUTPUT.PUT_LINE(v_last_name);
END;
```

9. In your own words, list the benefits of subprograms.

10. In your own words, describe what a stored procedure is.

11. The remaining questions in this practice use a copy of the employees table.

    A. Create the copy by executing the following SQL statement:

    ```
    CREATE TABLE employees_dup AS SELECT * from employees;
    ```

    B. Create the following procedure in Application Express:

    ```
    CREATE OR REPLACE PROCEDURE name_change IS
    BEGIN
      UPDATE employees_dup
        SET first_name = 'Susan'
        WHERE department_id = 60;
    END name_change;
    ```

    C. Save the definition of your procedure in case you need to modify it later. In the “Save SQL” popup, name your saved work “My name change procedure”.

    D. Execute the procedure by running the following anonymous block:

    ```
    BEGIN
      name_change;
    END;
    ```

12. SELECT from the table to check that the procedure has executed correctly and performed the UPDATE: Create a second procedure named pay_raise which changes the salary of all employees in employees_dup to a new value of 30000. Execute the procedure from anonymous block, then SELECT from the table to check that procedure has executed correctly.

13. Retrieve your first name_change procedure by clicking on its name in the Saved SQL window. Modify the code to remove OR REPLACE from the CREATE statement and change the department_id to 50. Execute your code to recreate the procedure. What happens?

14. This question uses the wf_countries table.

    A. Create a procedure that accepts a country_id as a parameter and displays the name of the country and its capitol city. Name your procedure get_country_info. Save your procedure definition for later use.

B. Execute your procedure from an anonymous block, using country_id 90.

C. Re-execute the procedure from the anonymous block, this time using country_id 95. What happens?

D. Retrieve your procedure code from Saved SQL and modify it to trap the NO_DATA_FOUND exception in an exception handler. Re-execute the procedure using country_id 95 again. Now what happens?