Do not award half marks.
In all cases give credit for appropriate alternative answers.

Question 1 (Compulsory) (30 marks)

(a) (i) In the context of an Oracle Database, what are *locks*? [2]

(ii) What are the **two** locking modes that Oracle uses in a multi-user database. How do these modes differ? [3]

(i) Locks are mechanisms that prevent destructive interaction between transactions (1 mark) that access the same resources (1 mark).

(ii) Exclusive (1 mark) and shared (1 mark). Shared locking accepts concurrent readers, whereas exclusive locking doesn't (1 mark).

(b) List the **three** sections of a PL/SQL block. [3]

Declaration (1 mark), executable (1 mark), and exception handling (1 mark).

(c) (i) In the context of PL/SQL, what is a *sequence*? [2]

(ii) What can indexes be used for? [1]

(i) A sequence is a database object created by a user (1 mark), and can be shared by multiple users/tables to generate unique integers (1 mark).

(ii) Sequences can be used to generate primary key values (1 mark).

(d) List **three** types of exception found in Oracle database. [3]

User-defined error (1 mark), predefined Oracle Server error (1 mark), and non-predefined Oracle server error (1 mark).

(e) In the context of PL/SQL, what are constraints? [2]

Constraints enforce rules at table level whenever a row is inserted, updated, ordeleted from the table (1 mark). The constraint must be satisfied for the operation to succeed (1 mark).

(f) Evaluate the following cursor declaration.

```
DECLARE
CURSOR emp_cursor IS
SELECT empno, ename, salary
FROM employee
WHERE empno=7654
FOR UPDATE of salary NOWAIT;
```

What does the last line of this cursor declaration do? [2]

This results in the cursor locking the affected row(s) (1 mark) when the cursor is opened (1 mark).

(g) Consider the tables provided at the end of this paper.

(i) Write a query to increase the salary of all employees working in department 10 by 10%. [3]

```
UPDATE      employee                    (1 mark)
SET         salary = salary*1.1         (1 mark)
WHERE       deptno = 10;                (1 mark)
```

(ii) Write a query to remove the records of all employees that were hired before February 16, 1981. [2]

```
DELETE (FROM) employee                  (1 mark)
WHERE hiredate < '16-FEB-81';           (1 mark)
```

(iii) Create a PL/SQL block to insert a new employee into the EMPLOYEE table, bearing in mind the following:

- use SQL*Plus substitution variables for the employee number, employment name and department number.
- hire date shall be the current date and time.
- leave the salary as null.

Make the changes permanent. Run the script as shown in the diagram below.

```
Enter the employment number: 7988
Enter the employment name: Jack
Enter the department number: 30

PL/SQL procedure successfully completed.
```

[5]

```
SET ECHO OFF
SET VERIFY OFF                          (1 mark)

ACCEPT p_empno PROMPT 'Enter the employee number:
'ACCEPT p_ename PROMPT 'Enter the employee name: 'ACCEPT
p_deptno PROMPT 'Enter the department number: '  (1 mark)
BEGIN
        INSERT INTO employee            (1 mark)
        VALUES (&p_empno, '&p_ename', sysdate, NULL,
        &p_deptno)                      (1 mark)
        COMMIT;                         (1 mark)
END;  /
```

(h) Define the term anonymous PL/SQL block. [2]

An anonymous PL/SQL block is an unnamed block (1 mark). It is declared at the point in an application where it is to be executed and is passed to the PL/SQL engine for execution at run-time (1 mark).

Question 2

(a) (i) In the context of Oracle Server, what are *privileges*? [1]

(ii) What are the *two* types of privilege? [2]

(iii) What privilege is required to log in to the Oracle Server? [1]

(iv) What does the WITH GRANT OPTION command do in the context of granting privileges? [1]

(i) Privileges are the right to execute particular SQL statements (1 mark)

(ii) System privileges (1 mark) and object privileges (1 mark)

(iii) The CREATE SESSION system privilege (1 mark)

(iv) It provides the user with the authority to pass on the privileges (1 mark)

(b) Assume that you are the owner of the EMPLOYEE table. Write an SQL statement to grant query privileges to user Scott on the EMPLOYEE table [3]

```
GRANT    select      (1 mark)
ON       employee    (1 mark)
TO       scott;      (1 mark)
```

(c) What is variable scope? [2]

Variable scope is the region of the program that can refer to the variable (1 mark). You can reference the declared variable within the executable section of that block and in all nested sub-blocks (1 mark).

(d) Consider the following PL/SQL block.

```
DECLARE
v_weight        NUMBER(3) :=600;
v_message       VARCHAR2(255) := 'Product 10012';
BEGIN
DECLARE
v_weight        NUMBER(3) :=1;
v_message       VARCHAR2(255) := 'Product 11001';
v_new_locn      VARCHAR2(50):= 'Europe';
BEGIN
v_weight := v_weight+1;
v_new_locn := 'Western' || v_new_locn;
END;
v_weight := v_weight+1;
v_message := v_message || ' is in stock';
v_new_locn := 'Western' || v_new_locn;

END;
```

Determine the value of each of the following variables according to the rules of scoping.

(i) The value of *v_weight* in the subblock. [1]

(ii) The value of *v_new_locn* in the subblock. [1]

(iii) The value of *v_weight* in the main block. [1]

(iv) The value of *v_message* in the main block. [1]

(v) The value of *v_new_locn* in the main block [1]

(i) 2
(ii) 'Western Europe'
(iii) 601
(iv) 'Product 10012 is in stock'
(v) illegal because v_new_locn is not visible outside in the subblock.

Question 3

(a) Define the following terms.

(i) Primary key constraint [2]

(ii) Check Constraint [1]

(iii) Referential integrity constraint. [2]

(i) The primary key constraint is a column or set of columns that uniquely identifies each row in a table (1 mark). This constraint enforces uniqueness of the column or column combination and ensures that no column that is part of the primary key can contain a null value (1 mark).

(ii) The check constraint defines a condition that each row must satisfy. (1 mark)

(iii) Also known as foreign key constraint (1 mark). It establishes and enforces a foreign key relationship between the column and a column of the referenced table (1 mark).

(b) Explain the CASCADE CONSTRAINTS clause used when dropping a table. [2]

The CASCADE CONSTRAINTS clause drops all referential integrity constraints (1 mark) that refer to the primary and unique keys defined on the dropped table (1 mark).

(c) Consider the following table instance chart:

Table name: EMPLOYEE

Column Name	**EMPNO**	**ENAME**	**HIREDATE**	**SALARY**	**DEPTNO**
Key Type	PK				FK
Null/Unique	NN, U	NN			NN
Default Value			System Date		
Check	1 to 9999				
FK Table					DEPARTMENT
FK Column					DEPTNO
Data Type	Number	Varchar2	Date	Number	Number
Length	4	10		7,2	3

Create the EMPLOYEE table based on the table instance chart shown above. Choose appropriate datatypes, and be sure to add integrity constraints. [6]

```
CREATE TABLE      employee                                    (1 mark)
      (empno      NUMBER(4)
      ename       VARCHAR2(10) NOT NULL
      hiredate    DATE DEFAULT SYSDATE
      salary      NUMBER(7,2),
      deptno      NUMBER(3) NOT NULL,                         (2 marks)

      CONSTRAINT  employee_empno_pk PRIMARY KEY (empno),      (1 mark)
      CONSTRAINT
            employee_empno_ck CHECK(empno between 1 and 9999),(1 mark)
      CONSTRAINT  employee_deptno_fk FOREIGN KEY(deptno)
      REFERENCES department(deptno));                         (1 mark)
```

(d) Create the EMPLOYEE2 table based on the structure of the EMPLOYEE table. Include only the EMPNO, ENAME and DEPTNO columns. Name the columns in your new table ID, LAST_NAME and DEPT_ID respectively. [2]

```
CREATE TABLE      employee2      AS                           (1 mark)
SELECT            empno id, ename last_name, deptno dept_id   (1 mark)
FROM              employee;
```

Question 4

(a) (i) What is a *view*? [1]

(ii) What are the tables on which views are based called? [1]

(iii) How are views defined using SQL? [1]

(i) A view is a logical table based on a table or another view (1 mark).

(ii) Base tables (1 mark).

(iii) In terms of SELECT statements (1 mark).

(b) (i) In the context of an Oracle Database, what is an *index*? [1]

(ii) Describe **two** ways of creating an index. [2]

(i) An index is an object that can speed up the retrieval of rows (1 mark).

(ii) A unique index can be created automatically be defining a PRIMARY KEY or UNIQUE constraint in a table definition (1 mark). Alternatively, users can create non-unique indexes on columns to speed up access time to rows (1 mark).

(c) Consider the tables provided at the end of this paper.

(i) Create a view called DEPT30_VU that contains the employee number, employee name, department number for all employees in department 30. Label the view columns EMPLOYEE_ID, EMPLOYEE and DEPARTMENT_NUMBER. Do not allow any DML operations on the view. [3]

(ii) Create a view called DEPT_SALARY_VU that contains the department name, minimum salary, maximum salary, and average salary for each department. [3]

(iii) Create an index on the foreign key column (deptno) in the EMPLOYEE table. [3]

(i)
```
CREATE VIEW dept30_vu
(employee_id, employee, department_number) AS          (1 mark)
SELECT      empno, ename, deptno
FROM        employee                                   (1 mark)
WITH READ ONLY;                                        (1 mark)
```

(ii)
```
CREATE VIEW dept_salary_vu AS                          (1 mark)
SELECT      dname, min(salary), max(salary), avg(salary) (1 mark)
FROM        employee, department
WHERE       employee.deptno=department.deptno
GROUP BY    dname;                                     (1 mark)
```

(iii)
```
CREATE INDEX                                           (1 mark)
employee_deptno_indx                                   (1 mark)
ON employee(deptno);                                   (1 mark)
```

Question 5

(a) List **three** types of PL/SQL LOOP control structures. [3]

Basic/Simple Loop (1 mark); FOR Loop (1 mark); and WHILE Loop (1 mark).

(b) Consider the following PL/SQL script.

```
SET SERVEROUTPUT ON
DECLARE
v_counter NUMBER :=5;
BEGIN
LOOP
DBMS_OUTPUT.PUT_LINE('v_counter = ' || v_counter);
v_counter:= v_counter-1;
IF v_counter =0
THEN
EXIT;
END IF;
END LOOP;
DBMS_OUTPUT.PUT_LINE('Done…..' );
END;
/
```

(i) What output will be produced following the execution of the script? [2]

(ii) How many times will the loop be executed? [1]

(iii) What is the EXIT condition for this loop? [1]

(i) **v_counter = 5**
v_counter = 4
v_counter = 3
v_counter = 2
v_counter = 1 **(1 mark)**
Done….. **(1 mark)**

(ii) **Five times** **(1 mark)**

(iii) **The EXIT condition for this loop is v_counter=0** **(1 mark)**

(c) Modify and rewrite the PL/SQL script shown in part b) using a WHILE loop. Make sure the output produced by this script does not differ from the output produced from the script shown in part b). [4]

```
SET SERVEROUTPUT ON
DECLARE
v_counter NUMBER :=5;                                          [1]
BEGIN
WHILE v_counter>0                                              [1]
LOOP                                                           [1]
DBMS_OUTPUT.PUT_LINE('v_counter = ' || v_counter);
v_counter:= v_counter-1;                                       [1]
END LOOP;
DBMS_OUTPUT.PUT_LINE('Done…..' );
END;
/
```

(d) Modify and rewrite the PL/SQL script shown in part b) using a FOR loop. Make sure the output produced by this script does not differ from the output produced from the script shown in part b). [4]

```
SET SERVEROUTPUT ON
BEGIN
FOR v_counter IN REVERSE 1 .. 5                          (2 marks)
LOOP                                                      (1 mark)
DBMS_OUTPUT.PUT_LINE('v_counter = ' || v_counter);
END LOOP;                                                 (1 mark)
DBMS_OUTPUT.PUT_LINE('Done…..' );
END;
/
```

- END OF PAPER -

EMPLOYEE Table

SQL> DESCRIBE employee:

Name	Null?	Type
EMPNO	NOT NULL	NUMBER(4)
ENAME	NOT NULL	VARCHAR2(30)
HIREDATE		DATE
SALARY		NUMBER(7,2)
DEPTNO	NOT NULL	NUMBER(3)

SQL> SELECT * FROM employee;

EMPNO	ENAME	HIREDATE	SALARY	DEPTNO
7839	KING	17-NOV-81	5000	10
7698	BLAKE	01-MAY-81	2850	30
7782	CLARK	09-JUN-81	2450	10
7566	JONES	02-APR-81	2975	20
7654	MARTIN	28-SEP-81	1250	30
7499	ALLEN	20-FEB-81	1600	30
7844	TURNER	08-SEP-81	1500	30
7900	JAMES	03-DEC-81	950	30
7521	WARD	22-FEB-81	1250	30
7902	FORD	03-DEC-81	3000	20
7369	SMITH	17-DEC-80	800	20
7788	SCOTT	09-DEC-82	3000	20
7876	ADAMS	12-JAN-83	1100	20
7934	MILLER	23-JAN-82	1300	10

DEPARTMENT Table

SQL> DESCRIBE department:

Name	Null?	Type
DEPTNO	NOT NULL	NUMBER (3)
DNAME		VARCHAR2 (14)
LOC		VARCHAR2 (13)

SQL> SELECT * FROM dept;

DEPTNO	DNAME	LOC
10	ACCOUNTING	NEW YORK
20	RESEARCH	DALLAS
30	SALES	CHICAGO
40	OPERATIONS	BOSTON