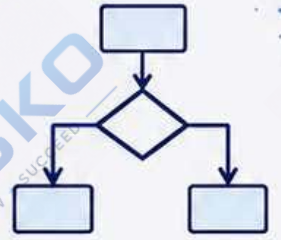


1010101010
0101010101
1010101010
0101010101



Notes





any 1 over ~
OR
any 2 out of 4 for 6 marks

Unit IV -

Q1.* What is Procedure? Explain steps to create procedure and Explain its advantages and disadvantages.



Procedure -

A procedure is a PL/SQL program that performs a task and does not return a value. You can call it whenever you need to perform a task.

Syntax -

```
CREATE OR REPLACE PROCEDURE
  procedure_name
```

```
(
  parameter1 datatype,
  parameter2 datatype
)
```

AS

BEGIN

-- statements

END;

A procedure in PL/SQL is a stored program that performs a specific task and does not return a value. You can call it whenever you need to perform a specific task.

Example - with Parameter

```
CREATE OR REPLACE PROCEDURE
```

```
show_name
```

```
(
```

```
    p_name varchar2 VARCHAR2
```

```
)
```

```
AS
```

```
BEGIN
```

```
    DBMS_OUTPUT.PUT_LINE ('Your name: ' || p_name);
```

```
END;
```

Call -

```
EXEC show_name ('Shaddu');
```

Example with out Parameter

```
CREATE OR REPLACE PROCEDURE
```

```
s_students
```

```
AS
```

```
BEGIN
```

```
    DBMS_OUTPUT.PUT_LINE ('Hello');
```

```
END;
```

Call -

```
EXEC s-student;
```

ProcedureFunction

- | | |
|--|--|
| 1) A procedure in PL/SQL is a stored program that performs a task and does not return any value. | A function in PL/SQL is a stored program that performs a specific task and returns a single value. |
| 2) In procedure RETURN keyword is optional and used only to exit. | RETURN keyword is mandatory. |
| 3) Procedure is called using EXEC or inside a block. | It can be called in SQL statements like SELECT. |
| 4) It cannot be used in SQL queries. | It can be used in SQL queries. |
| 5) A procedure allows both input and output parameters. | A function does not allow output parameters. |
| 6) Procedure can work with insert, update, delete, select statement. | Function only works with select statement. |
| 7) Procedure can use transactions. | Function cannot use transactions. |
| 8) We can call function from procedure. | We cannot call procedure from function. |

classmate

Date _____
Page _____

Advantages of Procedures -

1) Code Reusability -

You write the logic once, and use it many times by calling the procedure. This reduces repeated code in applications and makes development faster.

2. Better Performance -

Procedures are compiled and stored in the database.

When executed, the database runs the precompiled version which is faster than running SQL queries again and again.

3. Security & Access Control -

You can give users permission to run a procedure without giving them direct access to the tables.

This protects sensitive data and improves security.

4. Easy Maintenance

If something needs to change, you can edit the procedure at one place.

classmate

Date _____

Page _____

Q2. → What is function in PL/SQL?
Explain its structure and uses.

→

Function -

A function in PL/SQL is a stored program that performs a specific task and always returns a single value.

OR

A function is a named PL/SQL block that takes input, processes them, and return a value using RETURN statement.

Syntax -

CREATE OR REPLACE FUNCTION

function_name

(

parameter_name datatype

)

RETURN return_datatype

AS

-- variable declarations

BEGIN

-- statements

RETURN value;

EXCEPTION

-- error handling

END;

classmate

Date _____

Page _____

CREATE FUNCTION REPLACE FUNCTION
square_num

IN (Number) : RETURN NUMBER
AS BEGIN

RETURN num * num;
END;

Calling it -

SELECT square_num(5) FROM dual;

It returns square of 5 - 25

classmate

Date _____

Page _____

Eg-

CREATE OR REPLACE FUNCTION

add_num

(

a NUMBER,

b NUMBER

)

RETURN NUMBER

AS

result NUMBER;

BEGIN

result := a+b;

RETURN result;

EXCEPTION

WHEN OTHERS THEN

RETURN NULL;

END;

Calling it

SELECT add_num(10,20) FROM dual;

It returns addition of 10 and 20
30.

Q8. → Discuss the ACID properties of a transaction in detail with suitable examples.

→ transaction -

A transaction in a database is a group of operations that work together as a single unit.

To keep the data accurate and reliable every transaction must follow four properties known as ACID properties.

Automicity Consistency Isolation Durability.	{ These ACID properties ensure correctness, reliability and safe concurrent access in a database. }
---	--

1. Automicity -

Automicity means a transaction is treated as an indivisible unit.

All operations must succeed or none should take effect.

Eg -

In a money transfer, if amount is deducted from one account but cannot be added to the other, the entire transaction is rolled back.

2. Consistency -

Consistency ensures a transaction brings the database from one valid state to another valid state. All constraints and rules must be satisfied.

Eg -

A constraint that account balance cannot be negative ensures invalid transactions are rejected.

3. Isolation -

Isolation ensures that multiple transaction behaves as if it is the only one executing.

Eg -

When one user updates marks, another user cannot see partial updates until the transaction is committed.

4. Durability -

Durability guarantees that once a transaction is committed, its changes are permanent and will survive system failures.

Eg -

After booking a movie ticket, the confirmed booking remains even if system crashes.

Transaction

classmate

Date _____
Page _____

Backup & Recovery -

* Backup -

A backup is the copy of the database stored at a safe location.

It is used to restore data if something goes wrong, like system crash, disk failure or human errors.

Why Backup is Needed -

- Hardware or Disk Failure.
- Software crash.
- Accidental deletion or wrong update.
- Power failure or system crash.
- Natural disasters.

Types of Backups -

In the RDBMS, backups are classified based on what data is backed up and how often data is copied.

3 types

classmate

Date _____

Page _____

1. Full Backup -

- i) Full Backup completely copy the entire database.
- ii) It includes all data files, tables, indexes.
- iii) It takes more time and storage.
- iv) Full backup is the most reliable backup.

Uses When maximum safety is required.

2. Incremental Backup -

- i) Incremental backup only copies the data which is changed since last backup.
- ii) It is very fast and use less storage.
- iii) Its restoration is complex it need multiple backup.

Use:

When backup are taken frequently.

3. Differential Backup.

- i) This backup copies the data which is changed since the last full backup.
- ii) It is faster than full backup.

- iii) It is easier to recover than incremental

Use: Balance between speed and recovery time.

4. Logical Backup -

Logical Backup is a backup method in RDBMS where the database is saved in the form of logical SQL statements instead of physical data files.

It stores commands such as create, insert and alter that can recreate the database structure and data.

It backup the database objects like -

Tables, Views, Procedures, Stored SQL statements

Use:

For migration or structure-level backup

5. Physical Backup-

Physical backup is a type of backup method in RDBMS where the actual physical database files are copied and stored.

These files include data files, control files and log files exist on the disk.

When backup the database must usually be offline.

Use:

Quick disaster recovery.

* Recovery -

Recovery in RDBMS is the process of restoring the database to a correct and consistent state after a failure.

It ensures that all committed transactions are preserved and incomplete transactions are undo.

Need of Recovery -

- System crash or power failure
- Transaction failure
- Disk or Media failure
- Software and Hardware errors