Practical Prepration

(DDL) Data Definition Language

- `Create table`: to create a table in database
- `Alter table`: to modify the structure of table
- `Drop table`: to delete a table
- `Truncate table`: to delete all records from a table

(DML) Data Manipulation Language

- `Select`: to retrieve data from a database
- `Update` table: to modify the records in a table
- `Insert`: to insert new records into a database
- `Delete`: to delete records from a database

NOTE: Diffrence between `TRUNCATE` and `DELETE` is that `TRUNCATE` is a DDL command and `DELETE` is a DML command. Hence `TRUNCATE` can not be rolled back but `DELETE` can be rolled back.

(DCL) Data Control Language

- `Grant`: to allow specified users to perform specified tasks
- `Revoke`: to cancel previously granted or denied permissions
- `Rollback`: to undo a transaction
- `Commit`: to save a transaction

NOTE: `ROLLBACK` and `COMMIT` are transaction control commands as well as `DCL` commands.

Operators used with Select

- `DISTINCT`: to select unique records
- `ALIAS`: to rename a column or table using an alias name
- `WHERE`: to filter records
- `AND` and `OR`: to combine multiple conditions
- `ORDER BY`: to sort records in ascending or descending order
- `GROUP BY`: to group records with identical data
- `HAVING`: to filter out records that do not meet specified criteria
- `BETWEEN`: to filter records based on a range of values
- `IN`: to specify multiple possible values for a column
- `LIKE`: operator to search for a specified pattern in a column
- `NOT`: operator to negate a condition
- `UNION`: to combine the result-set of two or more SELECT statements
- `JOIN`: to combine records from two or more tables in a database
- `EXISTS`: to check the existence of any record in a subquery

Aggregate Functions

- `AVG()`: returns the average value
- `COUNT()`: returns the number of rows
- `FIRST()`: returns the first value
- `LAST()`: returns the last value
- `MAX()`: returns the largest value
- `MIN()`: returns the smallest value
- `SUM()`: returns the sum of the numeric values

PLSQL Commands

`PLSQL stands for Procedural Language extensions to SQL.`

Any PLSQL Code Block is divided into three sections:

- Declaration Section
- Execution Section
- Exception Section

Basic Syntax of PLSQL:

```
DECLARE
    -- Declaration Section
BEGIN
    -- Execution Section
EXCEPTION
    -- Exception Section
END;
```

Diffrent PLSQL Blocks are:

1. `POCEDURES`: It's a named block of statement. It may or may not return value. **Syntax:**

   ```
   CREATE [OR REPLACE] PROCEDURE procedure_name
       [ (parameter [IN | OUT | IN OUT] type [, ...])](IS | AS)
   BEGIN
       --Procedure Body
   END procedure_name;
   ```

2. `FUNCTIONS`: It's a named block of statement. It must return value. **Syntax:**

   ```
   CREATE [OR REPLACE] FUNCTION function_name
       [ (parameter [IN | OUT | IN OUT] type [, ...])](RETURN type |
   RETURN datatype)
   IS | AS
   BEGIN
       --Function Body
   END function_name;
   ```

3. TRIGGERS: It's a named block of statement. It's automatically fired or executed when DML statements like INSERT, UPDATE, DELETE are executed on a table or view. **Syntax:**

```
CREATE [OR REPLACE ] TRIGGER trigger_name
    {BEFORE | AFTER | INSTEAD OF }
    {INSERT [OR] | UPDATE [OR] | DELETE}
    [OF col_name]
    ON table_name
    [REFERENCING OLD AS o NEW AS n]
    [FOR EACH ROW]
DECLARE
    -- Variable and constant declarations
BEGIN
    -- Trigger Body
END;
```

4. CURSORS: It's a named private SQL area from where information can be retrieved. They are required to process rows individually for queries returning multiple rows. **Syntax:**

```
DECLARE
    -- Variable and constant declarations
BEGIN
    -- Cursor Body
END;
```