|  |
| --- |
| **Part A** |
| **Aim:** To create and drop View on the given table |
| **Prerequisite:** Relational Model |
| **Outcome:** Understanding and use of view. |
| **Theory:** A view is the tailored presentation of data contained in one or more table and can also be said as restricted view to the data in the tables. A view is a “virtual table” or a “stored query” which takes the output of a query and treats it as a table. The table upon which a view is created is called as base table .A view is a logical table based on a table or another view. A view contains no data of its own but is like a window through which data from tables can be viewed or changed. The tables on which a view is based are called base tables. The view is stored as a SELECT statement in the data dictionary .Advantages of a view:a. Additional level of table security.b. Hides data complexity.c. Simplifies the usage by combining multiple tables into a single tableSyntaxCreate or replace view view\_name AS SELECT column\_name(s) FROM table\_name WHERE condition Drop view ;ExampleCreate or replace view empview as select \* from emp;Drop view empview. |
| **Procedure:**   1. Formulate the query for given problem. 2. Write the SQL query with proper input. 3. Execute the query. |
| **Practice Exercise:**   1. Create the following tables for a Library Database:   BOOK (Book\_id, Bname, Publisher\_Name, Pub\_Year, Price, edition, ISBN)  Author(book number(10),author varchar(20),publisher varchar2(20),ISBN number(20));   1. Create a view PubDetails that display book name with publication details i.e. publisher name, publication year, edition, ISBN that are currently available in the Library. 2. Create a view that will display the book name with its author name. 3. Create a view from a view Pubdetails (created in question 2) which will display only book name and publisher name. 4. Drop all the views that are created. |
| **Instructions:**   1. Write and execute the query in Oracle SQL server. 2. Paste the snapshot of the output in input & output section. |
| **Part B** |
| **1.** |
| **2.** |
| **3.** |
| **4.** |
| **5.** |
| **Observation & Learning:**  From this experiment, we observed and learned how the following **SQL** commands to create and drop View on the given table are used to perform in the **ORACLE DATABASE** |
| **Conclusion:**  In this experiment, the following **SQL commands** are executed to to create and drop View on the given table in the **ORACLE DATABASE** and the Outputs are obtained as per queries. |
| **Questions:**   1. Explain materialized and non-materialized views? 2. **Materialized View -**A materialized view takes a different approach in which the query result is cached as a concrete table that may be updated from the original base tables from time to time. This enables much more efficient access, at the cost of some data being potentially out-of-date. It is most useful in data warehousing scenarios, where frequent queries of the actual base tables can be extremely expensive. **Non-Materialized View** - A Non-Materialized view takes the output of a query and makes it appear like a virtual table. You can use a view in most places where a table can be used. All operations performed on a view will affect data in the base table and so are subject to the integrity constraints and triggers of the base table. |