Cognizant_Logo_Brand_Blue_CMYK_300



­

Assignment Document:

<Redshift Advanced SQL>

Version: <Course Name>/ASSIGNMENT/xxxx/x.x

Date: 17-12-2019

Contents

[Topic: <REDSHIFT Advanced SQL> 3](#_Toc384718991)

[Hands-On Exercises 3](#_Toc384718992)

[Hands-On Exercise: <Creating view on REDSHIFT Tables > 3](#_Toc384718993)

[Guided Assignments 3](#_Toc384718996)

[Case Study Assignments 1](#_Toc384718997)0

[Case Study: <Application of views > 1](#_Toc384718998)0

Topic: <REDSHIFT Advanced SQL>

### Hands-On Exercises

Hands-On Exercise 1: <Creating view on REDSHIFT Tables >

Estimated Completion Time: xx Minutes

(xx Marks)

Objective: To Learn the multi set permanent and temporary table.

Complete the following assignment:

1. Guided Exercise 1
2. Guided Exercise 2
3. Guided Exercise 3

### 

### Guided Assignments

Guided Exercise 1: < Create a table “emp\_project” in database “ProjectXTables” . Create a view “Emp\_project\_view” on table emp\_project.

|  |
| --- |
| Problem Statement: |
| Create a permanent table “emp\_project” in database “ProjectXTables” with below specifications.  Columns of employee table:   1. emp\_id is an integer and define as distribution key. 2. Name is character datatype of length 30 bytes. 3. Joining\_location is character datatype of length 30 bytes. 4. Project\_location is character datatype of length 30 bytes 5. Salary is integer data type   Insert below records :   |  |  |  |  |  | | --- | --- | --- | --- | --- | | Emp\_id | Name | Joining\_location | project\_location | salary | | 1001 | Ravi | Hyd | Ban | 5000 | | 1005 | Seshu | Ban | Ban | 4000 | | 1002 | John | Kol | Pun | 6500 | | 1003 | Mona | Pun | Hyd | 4000 |     Create a view “Emp\_project\_view” on table emp\_project which should display the eligible bonus of each employee as per below specifications:   * + If project location and joining location is same then bonus should be 10% of salary .   + If project location and joining location is different , then bonus should be 20% of salary. |

Estimated Completion Time: 10 Minutes

(xx Marks)

Objective: To learn the creation of view

Concept: In REDSHIFT, views will restrict the users to access the columns of a table. Views will allow to derive the columns based on the existing data.

|  |
| --- |
| Solution |
| Step 1: Open SQL Interface.  Step 2: Logon to REDSHIFT server with credentials.  Step 3 : To create a table in a particular database, set your default database with the below statement.  Database ProjectXtables;  Step 4 :  CREATE TABLE emp\_project  ( emp\_Id INTEGER,  name varchar(30),  joining\_location varchar(30),  project\_location varchar(30),  salary integer  ) distkey (emp\_Id);  Step 5 :  Insert the below the records.  Insert into emp\_project (1001,’Ravi’,’Hyd’,’Ban’,5000);  Insert into emp\_project (1005,’Seshu’,’Ban’,’Ban’,4000);  Insert into emp\_project (1002,’John’,’Kol’,’Pun’,6500);  Insert into emp\_project (1003,’Mona’,’Pun’,’Hyd’,4000);  Step 6: create view on top of the base table:  Create view emp\_project \_view AS  Select  emp\_Id,  name,  joining\_location,  project\_location,  salary ,  case when trim(joining\_location) = trim(project\_location)  then (salary \* 10/100) else (salary \* 20/100) end as bonus  from ProjectXtables. emp\_project with no schema binding;  Step 7 : Run the view to display the result:  Select \* from emp\_project \_view; |

Summary of this exercise:

You have just learnt:

* Create or replace view statement with multi set property.

Guided Exercise 2: < Create a table “student\_marks” in database “ProjectXTables” . Create a view “student\_marks\_view” on table student\_marks.

|  |
| --- |
| Problem Statement: |
| Create a table “student\_marks” in database “ProjectXTables” with below data specifications.  Insert below records :     |  |  |  |  |  | | --- | --- | --- | --- | --- | | Student\_no | name | Maths\_marks | Physics\_marks | Chemistry\_marks | | 100 | Ravi | 54 | 87 | ? | | 101 | Venu | 83 | ? | ? | | 103 | Raju | 64 | 36 | 27 |   Create a view “student\_marks\_view” on table student\_marks which should display the data as shown below:   |  |  |  |  | | --- | --- | --- | --- | | Student\_no | Name | Subject\_name | marks | | 100 | Ravi | Maths | 54 | | 100 | Ravi | Physics | 87 | | 100 | Ravi | Chemistry | ? | | 101 | Venu | Maths | 83 | | 101 | Venu | Physics | ? | | 101 | venu | Chemistry | ? | | 103 | Raju | Maths | 64 | | 103 | Raju | Physics | 36 | | 103 | Raju | chemistry | 27 | |

Estimated Completion Time: 10 Minutes

(xx Marks)

Objective: To learn the creation of view

Concept: In REDSHIFT, views will restrict the users to access the columns of a table. Views will be used to generate the report in the required format.

|  |
| --- |
| Solution |
| Step 1: Open SQL Interface.  Step 2: Logon to REDSHIFT server with credentials.  Step 3 : To create a table in a particular database, set your default database with the below statement.  Database ProjectXtables;  Step 4 :  CREATE TABLE student\_marks  ( student\_no INT,  name varchar(30),  Maths\_marks int,  Physics\_marks int,  Chemistry\_marks int  ) diststyle all;  Step 5 :  Insert the below the records.  Insert into student\_marks (100,’Ravi’,54,87,null);  Insert into student\_marks (101,’Venu’,83,null,null);  Insert into student\_marks (103,’Raju’,64,36,27);  Step 6: Create a view which should display the base table data in the given format:  Create view student\_marks\_view as  select student\_no , name , ‘Maths’ as subject\_name , Maths\_marks as marks  from student  union all  select student\_no , name , ‘Physics’ as subject\_name , Physics\_marks as marks  from student  union all  select student\_no , name , ‘Chemistry’ as subject\_name , Chemistry\_marks as marks  from student with no schema binding ;  Step 7: Run the view to display the result:  Select \* from student\_marks\_view; |

Summary of this exercise:

You have just learnt:

* Create or replace view statement with operator Union all.

Guided Exercise 3: < Create a table “std\_marks\_detail” in database “ProjectXTables” . Create a view “student\_marks\_view” on table std\_marks\_detail.

|  |
| --- |
| Problem Statement: |
| Create a table “std\_marks\_detail” in database “ProjectXTables” with below data specifications.  Insert below records :     |  |  |  |  | | --- | --- | --- | --- | | Student\_no | Name | Subject\_name | marks | | 100 | Ravi | Maths | 54 | | 100 | Ravi | Physics | 87 | | 100 | Ravi | Chemistry | ? | | 101 | Venu | Maths | 83 | | 101 | Venu | Physics | ? | | 101 | venu | Chemistry | ? | | 103 | Raju | Maths | 64 | | 103 | Raju | Physics | 36 | | 103 | Raju | chemistry | 27 |   Create a view “student\_marks\_view” on table std\_marks\_detail which should display the data as shown below:   |  |  |  |  |  | | --- | --- | --- | --- | --- | | Student\_no | name | Maths\_marks | Physics\_marks | Chemistry\_marks | | 100 | Ravi | 54 | 87 | ? | | 101 | Venu | 83 | ? | ? | | 103 | Raju | 64 | 36 | 27 | |

Estimated Completion Time: 10 Minutes

(xx Marks)

Objective: To learn the creation of view

Concept: In REDSHIFT, views will restrict the users to access the columns of a table. Views will be used to generate the report in the required format.

|  |
| --- |
| Solution |
| Step 1: Open SQL Interface.  Step 2: Logon to REDSHIFT server with credentials.  Step 3 : To create a table in a particular database, set your default database with the below statement.  Database ProjectXtables;  Step 4 :  CREATE TABLE std\_marks\_detail  ( student\_no INT,  name varchar(30),  subject\_name varchar(30),  marks integer  ) diststyle even;  Step 5 :  Insert the below the records.  Insert into std\_marks\_detail (100,’Ravi’,’Maths’,54);  Insert into std\_marks\_detail (100,’Ravi’,’Physics’,87);  Insert into std\_marks\_detail (100,’Ravi’,’Chemistry’,null);  Insert into std\_marks\_detail (101,’Venu’,’Maths’,83);  Insert into std\_marks\_detail (101,’Venu’,’Physics’,null);  Insert into std\_marks\_detail (101,’Venu’,’Chemistry’,null);  Insert into std\_marks\_detail (103,’Raju’,’Maths’,64);  Insert into std\_marks\_detail (103,’Raju’,’Physics’,36);  Insert into std\_marks\_detail (103,’Raju’,’Chemistry’,27);  Step 6: Create a view which should display the base table data in the given format:  Create view student\_marks\_view as  (  select  student\_no , name , max( maths\_marks) ,max( physics\_marks ),max( chemistry\_marks)  From (  select student.student\_no , student.name ,  case when subject\_name = ‘Maths’  then marks  else null end as maths\_marks,  case when subject\_name = ‘Physics’  then marks  else null end as Physics\_marks,  case when subject\_name = ‘Chemistry’  then marks  else null end as Chemistry\_marks,  from student ,  (select student\_no, name  from student group by student\_no,name) derived\_table1  where derived\_table1.student\_no = student.student.student\_no ) derived\_table2  Group by student\_no,name  ) with no schema binding;  Step 7: Run the view to display the result:  Exec student\_marks\_view ; |

Summary of this exercise:

You have just learnt:

* Create view statement with derived table specification.

### Case Study Assignments

Case Study: <Application of Sub Query >

Estimated Completion Time: xx Minutes

(xx Marks)

Objective: To manipulate the data with the help of views.

Case: Create the below table with given specifications.

|  |
| --- |
| Problem Statement |
| Database name is : ProjectXtables  Table name is : emp\_loc  Column Specification:  Table Specification :   1. Emp\_no : defines the employee number and its data type is integer. 2. Name : describes the name of the employee and data type is varchar(30) 3. Preferred\_loc : describes the preference locations of employee. A employee can prefer multiple locations.   Sample records are :   |  |  |  | | --- | --- | --- | | Emp\_no | Name | Preferred\_loc | | 100 | Raju | Hyderabad | | 100 | Raju | Pune | | 101 | Tina | Chennai | | 101 | Tina | Bangalore | | 101 | Tina | Kolkata | | 102 | Joshi | Kolkata | | 103 | Ashish | Pune | | 103 | Ashish | Kolkata |   Employer will choose one of the preferred locations of each employee based on priority table of employer. The preferred location list for employer is as below:   |  |  | | --- | --- | | Location | preference | | Hyderabad | First | | Chennai | second | | Pune | third | | Bangalore | four | | Kolkata | last |     Suppose an employee ‘A’ chooses two locations like ‘Hyderabad’ and ‘Pune’, then employer will choose ‘Hyderabad’ for ‘A’ . Because, the employer has ‘Hyderabad’ with first preference and ‘Pune’ with third preference.   * Write a sql which will identify first preferred location of the employee as per employer preference list and load the data into employer\_prefer table.   employer\_prefer table column details:   |  |  |  | | --- | --- | --- | | Emp\_no | Name | First\_Preferred\_loc | | 100 | Raju | Hyderabad | | 101 | Tina | Chennai | | 102 | Joshi | Kolkata | | 103 | Ashish | Pune | |

|  |
| --- |
| Solution: |
| Step 1 : Create the table emp\_loc in “ProjectXTables ” database with the given specifications and insert the records into the table:  CREATE TABLE ProjectXTables . emp\_loc  (  Emp\_no integer,  Name varchar(30),  Preferred\_loc varchar(30)  distkey (emp\_no) ;  Step 2 : Insert the below records into emp\_loc table:  Insert into emp\_loc (100,’Raju’,’Hyderabad’);  Insert into emp\_loc (100,’Raju’,’Pune’);  Insert into emp\_loc (100,’Raju’,’Hyderabad’);  Insert into emp\_loc (101,’Tina’,’Chennai’);  Insert into emp\_loc (101,’Tina’,’Bangalore’);  Insert into emp\_loc (101,’Tina’,’Kolkata’);  Insert into emp\_loc (102,’Joshi’,’Kolkata’);  Insert into emp\_loc (103,’Ashish’,’Pune’);  Insert into emp\_loc (103,’Ashish’,’Kolkata’);  Step 3: Create target table employer\_prefer:  CREATE TABLE ProjectXTables . employer\_prefer  (  Emp\_no integer,  Name varchar(30),  First\_Preferred\_loc varchar(30)  distkey (emp\_no) ;  Step 4: Create a sql which will identify the preferred location and load into target table:  Insert into ProjectXTables . employer\_prefer  Select  Empno , Ename,  Case when priority\_loc = 1 then ‘Hyderabad’  when priority\_loc = 2 then ‘Chennai’  when priority\_loc = 3 then ‘Pune’  when priority\_loc = 4 then ‘Bangalore’  when priority\_loc = 5 then ‘Kolkata’ End as first\_ Preferred\_loc  From  ( select Empno,  Ename,  Min (  Case when Preferred\_loc =‘Hyderabad’ then 1  when preferred\_loc =‘Chennai’ then 2  when preferred\_loc = ‘Pune’ then 3  when preferred\_loc = ‘Bangalore’ then 4  when preferred\_loc = ‘Kolkata’ then 5  End ) As priority\_loc  From  ProjectXTables .Emp\_loc  Group by Empno, Ename  ) AS derived\_table1;  END;  Step 5:Run the sql and populate the target table  Step 6: Check the target table data:  Select \* from ProjectXTables . employer\_prefer; |

Evaluation Rubrics

|  |  |
| --- | --- |
| Parameters | Weightage |
| 1. Completeness | X |
| 1. Accuracy | X |
| 1. Clarity of understanding | X |
| 1. Presentation | X |
| Total |  |

Summary of this Case Study:

You have just learnt:

* Creation of stored procedure.
* Usage of case construct.
* Usage of derived table.