**Ex 11**

**Task 1 Verify Creating a Denormalized Table.**

**A picture containing table

Description automatically generated**

**Task 2 Retrieve Data to Fill the Denormalized Table.**

**Graphical user interface, text, application

Description automatically generated**

**Task 3 Modify the Task 2 Query to Eliminate Incomplete Data.**

**Graphical user interface, text, application

Description automatically generated**

**Task 4 Answer the Question**

**Text, letter

Description automatically generated**

Answer: The data that does not get selected in the Task 3 query is the data with no business purpose. Essentially, things that are not purchased and not owned by the company. Using the full join on inventory and asset\_desc was getting asset descriptions that are not in inventory. By changing the outer join of inventory to asset\_desc to an inner join eliminates this data due to the requirements set in the query.

**Task 5 Create a View from the Task 3 Query.**

**Graphical user interface, text, application, email

Description automatically generated**

Note: I thought that it was very interesting that my text\_length did not match your text\_length. I did this assignment twice, once for the IC and once for the Ex11 assignment. While completing this task for the IC’s my text\_length differed. I must have taken a few spaces out while selecting the data.

**Task 6 Insert Data into the Denormalized Table.**

**Graphical user interface, text, application

Description automatically generated**

**Task 7 Verify Comparable Indexes.**

**Graphical user interface, text, application, email

Description automatically generated**

**Task 8 Run a Query against the Normalized Tables.**

**Graphical user interface, text, application, email

Description automatically generated**

**Re-run as Normalized:**

**Text

Description automatically generated**

**Task 9 Run a Query against the Denormalized Tables.**

**Graphical user interface, text, application

Description automatically generated**

**Re-run as Denormalized:**

**Graphical user interface, text, application

Description automatically generated**

**Task 10 Explain the Plan for the Task 8 Query on the Normalized Tables.**

**Graphical user interface, text, application

Description automatically generated**

**Task 10 Part 2. Once the plan has been explained, retrieve the plan data from the buffer where it is saved (temporarily).Graphical user interface, application, table

Description automatically generatedGraphical user interface, application

Description automatically generated with medium confidence**

**Task 11 Explain the Plan for the Task 9 Query on the Denormalized Tables.**

**Graphical user interface, text, application, email

Description automatically generated**

**Task 11 Part 2. Once the plan has been explained, retrieve the plan data from the buffer where it is saved (temporarily).**

**Graphical user interface, table

Description automatically generatedTable

Description automatically generated**

**Task 12 Verify Privileges. Logged into the class server, verify you have been granted access to the v$sql system view.**

**Graphical user interface, text, application

Description automatically generated**

**Task 13 CPU Time for the Task 8 Query.**

**Graphical user interface, text, application

Description automatically generated**

**Task 14 CPU Time for the Task 8 and Task 9 Query.**

**Text, application

Description automatically generated**

**Task 15 Compare Denormalized to Normalized.**

**Text, letter

Description automatically generated**

**Answer:** In the task 10 query we can see that the explanation for the normalized

query takes 10 steps. Using the explanation for the denormalized query in task

11 we can see that the denormalized query takes 4 steps. Using these results

alone we could hypothesize that the denormalized query in task 11 will perform

better than the normalized query used in task 10. However, using the results

from task 14 we can see that the denormalized query does perform better. The

results show that the denormalized query processed the same number of rows

with a faster CPU\_TIME and a faster ELAPSED\_TIME. In conclusion, it appears

that denormalization does improve the performance of the query.