**LAB # 03**

**Objectives :** To learn query execution in OLTP databases

**LAB TASK:**

Execute OLTP queries for the following scenario:

*An organization has a main data table that lists events. Each event has a series of fields including duration, reason code and status.*

*The OLTP system has a table of statuses with attributes status code and Name and there is a common field between the main table (status code), which allows the organization to print the status name rather than the code.*

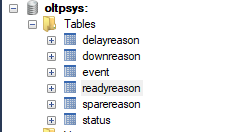
*There are four statuses (ready, delay, spare and down) and there are a range of reasons associated with each status with reason codes. It is possible for the same reason code to exist in two statuses (e.g. delay maintenance and down maintenance).*

*For some reason, the source database has four tables of reasons (one for each status). Each of these has a field which links to the main reason code.*

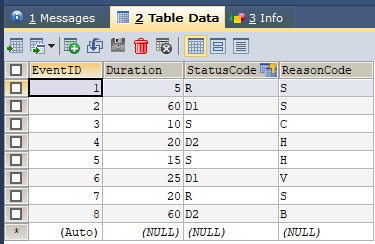
*The organization wishes to do the following:*

*When the main table record status is ‘delay’, then retrieve the name in the ‘delay’ table which corresponds to the main table delay code. When the status is ‘down’, do the same thing but look up the name in the ‘down’ table.*

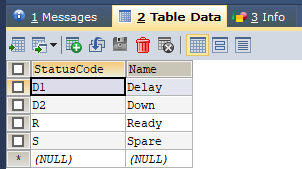
***RESULTS:***



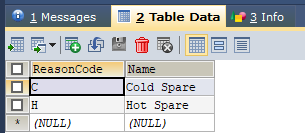
**Event:**



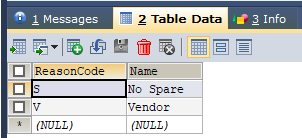
**Status:**



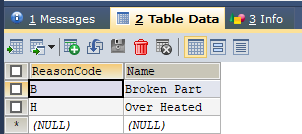
**SpareReason:**



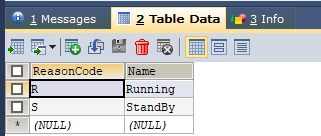
**DelayReason:**



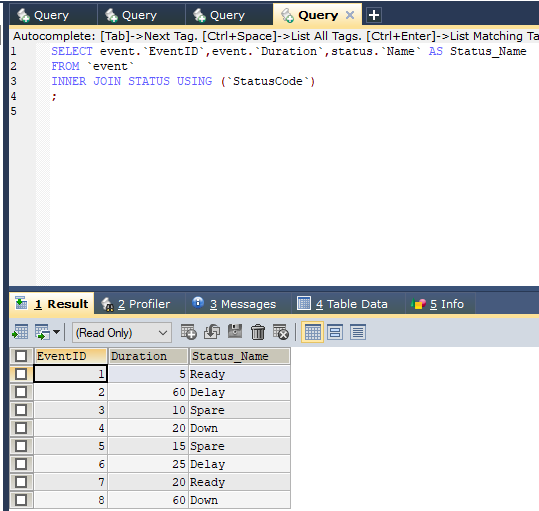
**DownReason:**



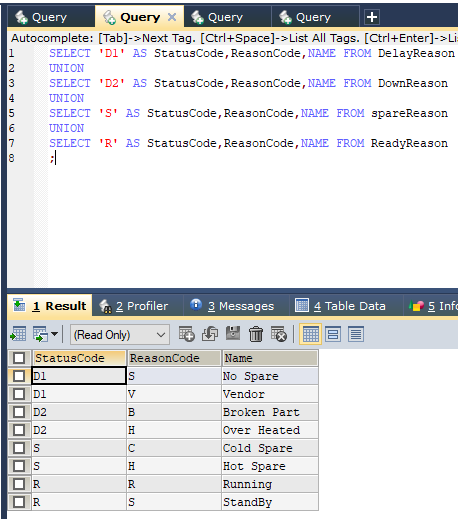
**ReadyReason:**



TheFirst thing that you have to retrieve is the Status name using their respective Status Code in the `oltp:`.Event table in order to do this we are using INNER JOIN OPERATOR.



The Second thing that we have to do is retrieve Reason Name of their appropriate Reason Code that are relates to their respective Status Code. In oder to do this we are using UNION OPERATOR that union all the result of different reason table and put them in a result set named Reason that is used for matching the result of the inner join conditions.



Now we merge both these results into a single query

