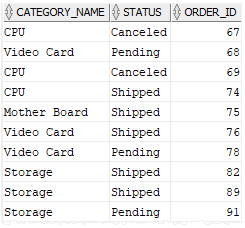
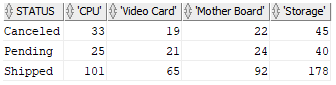
You will always need an aggregate function in the pivot. If you wanted to transpose rows to columns without losing the detail in your data, you could try a MAX or MIN function(see final example).

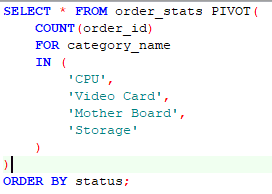
A simple example would be



To be converted to

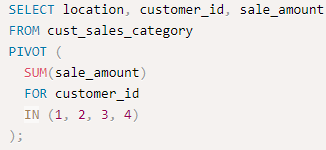


Using,

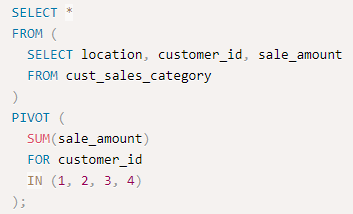


You basically, write you select in this case “select \* from order\_stats” then you say “PIVOT(“ aggregate function for <column name> in ( your pivot criteria ) remember this has to be select \* and cannot be select column1, column2,column3… cause it throws ORA-00904: " ": invalid identifier

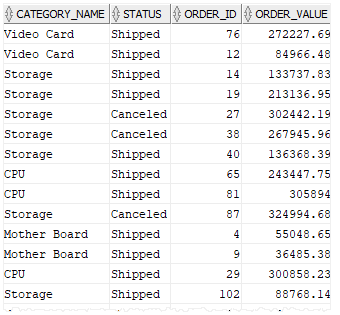
error, in such a cases you need to do a “select \* from ( select column1, column2”. Below won’t work



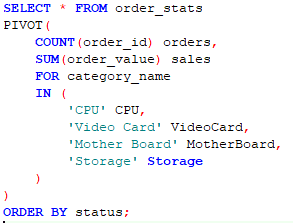
So modify it as below

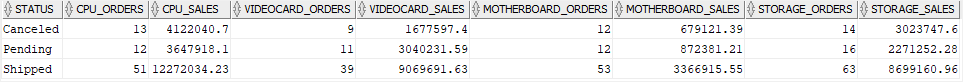


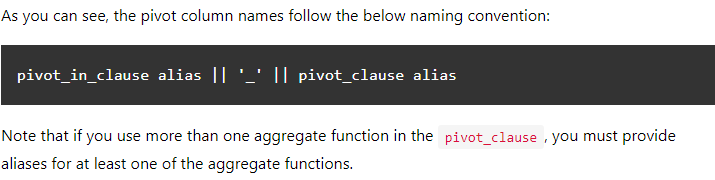
When things get complicated i.e you need to aggregate and pivot on multiple columns eg count and sum for some data that looks like this.



To look like this,



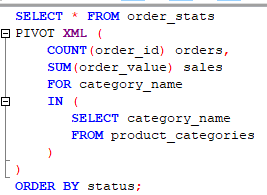




Oracle PIVOT with subquery

You cannot use a [subquery](https://www.oracletutorial.com/oracle-basics/oracle-subquery/) in the pivot\_in\_clause

This restriction is relaxed with the XML option:



**As an example below query will give you the time taken by a particular request every hour from the request log table.**

**select** \* **from** (

**select**

**extract**(**day** **from** start\_time) **as** DOM

,**extract**(**hour** **from** start\_time) **as** HOD

,**to\_char**(**round**(**sum**( **extract**(**minute** **from** end\_time-start\_time)\*60 + **trunc**(**extract**(**second** **from** end\_time-start\_time)))/**count**(\*),2)) **as** avg\_secs

**from** requests\_log\_t a **where**

method='RESEAU.STATUS.GET' **and**

**trunc**(start\_time) > **sysdate**-10

**group** **by**

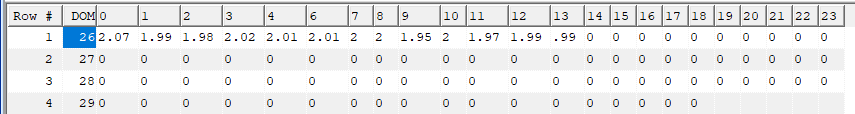
**extract**(**day** **from** start\_time)

,**extract**(**hour** **from** start\_time)

)

PIVOT

(**min**(avg\_secs) **for** HOD **in** ( 0, 1, 2,3,4,6,7,8,9,10,11,12,13,14,15,16,17,18,19,20,21,22,23))



**However the problem is if you want the number of occurrences and the total time taken it would be difficult as multiple aggregations will be needed.So this is what I did, maybe there is a better way.**

**select** \* **from** (

**select**

**extract**(**day** **from** start\_time) **as** DOM

,**extract**(**hour** **from** start\_time) **as** HOD

,**to\_char**(**nvl**(**sum**(**extract**(**minute** **from** end\_time-start\_time)\*60+ **trunc**(**extract**(**second** **from** end\_time-start\_time))),0))

|| ' Secs/'

|| **count**(\*)

|| ' = '

|| **to\_char**(**round**(**sum**( **extract**(**minute** **from** end\_time-start\_time)\*60 + **trunc**(**extract**(**second** **from** end\_time-start\_time)))/**count**(\*),2))

|| ' Secs' **as** avg\_secs

**from** requests\_log\_t a **where**

method='RESEAU.STATUS.GET' **and**

**trunc**(start\_time) > **sysdate**-10

**group** **by**

**extract**(**day** **from** start\_time)

,**extract**(**hour** **from** start\_time)

)

PIVOT

(**min**(avg\_secs) **for** HOD **in** ( 0, 1, 2,3,4,6,7,8,9,10,11,12,13,14,15,16,17,18,19,20,21,22,23))

