Andrew Ho <kironide@gmail.com>

SQL 03/12 Summary & Notes

1 message

Jonah & Robert Signal Data Science <signaldatascience@gmail.com> Mon, Mar 14, 2016 at 12:31 AM
To: Ali Bagherpour <ali.bagherp@gmail.com>, Andrew Ho <Kironide@gmail.com>, Chad Groft <clgroft@gmail.com>, David Bolin <david@bolin.at>, Jacob Pekarek <jpekarek@trinity.edu>, Jaiwithani <jaiwithani@gmail.com>, James Cook <cookjw@gmail.com>, Jonah & Robert Signal Data Science <signaldatascience@gmail.com>, Linchuan Zhang <email.linch@gmail.com>, Matthew Gentzel <magw6270@terpmail.umd.edu>, Olivia Schaefer <taygetea@gmail.com>, Robert Cordwell <cordwell@gmail.com>, Sam Eisenstat <sam.eisenst@gmail.com>, Tom Guo <tomguo4@gmail.com>, Trevor Murphy <trevor.m.murphy@gmail.com>

Hi All,

SQL is both quick to learn and incredibly important for interviews. A common mistake is to assume that, because it's relatively simple compared to many other data science topics, that it's less impressive to know.

Interview Questions

Blue Group Questions
Red Group Questions
Green Group Questions
Yellow Group Questions

Please feel free to edit these! Some of you had very clever solutions and comments, and I'd love to see that knowledge and skill written down.

Additional topics for self-study
Coalesce(thing1, thing2, ..., lastthing) : equivalent to case when thing1 is not null then thing1 when thing2 is not null then thing2 when ... else lastthing end

Union all combines two tables vertically (tables must have columns with the same name and same data type). Union does the same thing but ensures distinct rows.

Partitions! Useful when you want to compare analytic functions over different spaces. Know your basic sum(x) over (partition by y, z)

Window framing: Full-spectrum control over partitions. Learn this and use it to achieve God Tier SQL mastery

Additional tips and tricks

Write queries inside out. What would be top-to-bottom in another programming language is inner-to-outer in SQL.

Complex condition on which rows to choose? Use a subquery. If you have something of the form "where X in (select x from ...) then the "(select x from ...)" does not need a table name.

Counting # of times something is true:
sum(case when "X is true" then 1 else 0 end)

Finding a distribution of the number of times something appears:

```
select hits, count(*)
from (
  select x, count(*) as hits
  from table
  group by 1
```

```
   ) x
group by 1
```

Getting two "summary statistics" and combining them together

```
select value_x, value_y
from (
  select Value X as value_x
  from Table
  ) a
join (
  select Value Y as value_y
  from Table
  )b
```

Most joins are inner joins, then left/right joins. Outer joins are unusual, and cross joins (which give you the cross project of all valid rows with each other) are so rare you'll know you need one when you need to explode the # of rows.

Best,
Robert