We can then easily find the min/max dates in each group

```
with a  as (
(select fail_date as date,
       'failed' as period_state
       from failed)
union all

 (select success_date as date,
        'succeeded' as period_state
        from succeeded)
    ),

  b as (
select date,
       period_state,
       row_number() over (order by period_state, date asc) as seq
   from a where date between '2019-01-01' and '2019-12-31'
         ),

 c as (
select date, period_state,seq, dateadd(d, -seq, date) as seqStart from b
)

select period_state, min(date) as start_date, max(date) as end_date from c
group by seqStart,period_state
order by start_date asc
```


Type comment here... (Markdown is supported)

Post

killcars ★ 115 October 27, 2019 3:34 PM

We don't need `c` , we can directly get result from `b`

```
with a  as (
(select fail_date as date,
       'failed' as period_state
       from failed)
union all
 (select success_date as date,
        'succeeded' as period_state
        from succeeded)
    ),
```

Read More

8 Show 1 reply Reply

qlcheng ★ 6 November 4, 2019 8:33 AM

wow...honestly...I don't think I would be able to come up with this solution without dedicated 5+ years of SQL experience

3 Reply

dixit09 ★ 7 March 14, 2020 4:38 AM

using row_num to generate groups

```
with cte as
(
    select fail_date dt, 'failed' as status
    from Failed
    union all
    select success_date dt , 'succeeded' as status
    from Succeeded
), rn_cte as
(select dt, status,
```

Read More

0 Reply

yuting1013 ★ 0 January 3, 2020 10:19 PM

quite good!!

0 Reply