

# Week3

## Week 3

- TLE refers to Time Limit Extended

[Week 3](#)

[Meta data about posts](#)

[Compare Posts with PostsWithDeleted](#)

[Try to implement 'TRIM' function](#)

[Join](#)

[Common](#)

[Inner Join](#)

[Left Join](#)

[Left Join Update](#)

[Gather Data](#)

[Save 'JS' to increase speed](#)

[Finally Works](#)

[Try to add TRIM](#)

[Differencr between TRIM](#)

## Meta data about posts

Database Schema

Posts

Total items: 41458190

Id,Body,Title,Tags

PostsWithDeleted

Total items: 51653488

Id,Body,Title,Tags

PostHistory(After viewing,maybe not relavent)

Total items: 108471877

Id,PostId,Comment,Text

## Compare Posts with PostsWithDeleted

```
select * from PostsWithDeleted
except
select * from Posts
```

TLE

```
(select Top 5000 * from PostsWithDeleted pd )
except
(select Top 5000 * from Posts p )
```

52.9

ms

52.8

ms

53.1

ms

23.4

ms

Results

Messages

Download CSV

id	CreationDate -	DeletionDate -	Score -	View...	Body -	Own...	Own...	Last...	Last...	LastEditDate -	LastActivityDate -	Title -	Tags -	Ans...	Com...	Favo...	ClosedDate -	CommunityOwnedDate
	2008-07-31 21:26:37	2011-03-28 00:53:47	1										data					
1	2008-07-31 21:27:50	2008-08-01 00:46:53	1															
1	2008-07-31 21:29:54	2008-08-01 00:46:53	1															
4	2008-07-31 21:46:20	2009-08-01 23:27:48	10															
	2008-07-31 23:33:19	2015-02-11 08:26:40	42									c# code-generation	js visualjs				2013-06-03 04:00:25	
9	2008-07-31 23:42:31	2010-01-25 20:05:29	19															
14	2008-08-01 01:01:10	2009-01-11 07:20:36	-15															
8	2008-08-01 07:25:22	2013-05-28 11:45:29	4															
	2008-08-01 12:09:41	2012-12-13 18:04:53	25										php				2012-05-18 11:12:42	
13	2008-08-01 12:17:41	2010-11-08 17:54:41	-3															
25	2008-08-01 12:23:39	2011-06-28 18:19:50	1															
9	2008-08-01 12:32:48	2011-02-03 20:14:31	2															
25	2008-08-01 12:36:00	2017-07-03 07:59:56	34															
	2008-08-01 12:43:11	2008-08-01 13:18:31	0										c# vb.net timer					
	2008-08-01 12:44:02	2009-08-24 01:44:32	0										c# inversion-of-control structuremap alt...					
23	2008-08-01 12:49:12	2012-12-13 18:04:53	8															

2911 rows returned in 74 ms

### Solution

Just use Post.

## Try to implement 'TRIM' function

```
select * from (
SELECT REPLACE(
    REPLACE(
        REPLACE(
            LTRIM(RTRIM(Body))
            , ' ', ' '+CHAR(7)) --Changes 2 spaces to the OX model
        ,CHAR(7)+' ', '') --Changes the XO model to nothing
    ,CHAR(7), '') AS p --Changes the remaining X's to nothing
FROM Posts pt) p
```

```
select * from (
SELECT REPLACE(
    REPLACE(
        REPLACE(
            REPLACE(
                LTRIM(RTRIM(Body))
                , '\n', '')
            , ' ', '')
        , '\n', '')
        , '\t', '') AS p
FROM Posts pt) p
```

```
select Body from (
SELECT REPLACE(LTRIM(RTRIM(Body)),' ','') as Body
FROM Posts pt) p
```

## Join

## Common

```
SELECT TOP 1000 c.PostId AS [Post Link], YEAR(C.CreationDate) AS Year,
Month(C.CreationDate) AS Month, c.text AS Body
FROM Comments C, Posts P
```

```
Where (
  C.Text LIKE '%bastard%' AND
  C.PostID = P.ID AND
  P.ClosedDate IS NULL
)
Order By Year, Month
```

## Inner Join

```
select Top 100 p.Id,c.PostId
from Posts p inner join Comments c
on p.Id = c.PostId
order by p.id
```

## Left Join

```
select Top 100 p.Id,c.PostId
from Posts p left join Comments c
on p.Id = c.PostId
where p.Body like '###key1###'
order by p.id
```

## Timeout Expired

## Left Join Update

```
select count(*)
from Posts p left join Comments c
on p.Id = c.PostId
```

## Gather Data

```
select count(*)
from Posts p left join Comments c
on p.Id = c.PostId
```

result:87141992

```
select count(*)
from Posts p left join Comments c
on p.Id = c.PostId
where
(p.Title like '###key###' or p.Title like '###Method###') and
p.Body like '###key###' and p.Body like '###Method###' and
( p.Body like '%js%'or p.Body like '%JS%')
```

```
select count(*)
```

```

from Posts p left join Comments c
on p.Id = c.PostId
where
(p.Title like '###key###' or p.Title like '###Method###') and
(
(p.Body like '###key###' and p.Body like '###Method###') or
(c.Text like '###key###' and c.Text like '###Method###')
) and
(( p.Body like '%js%')
or
(c.Text like '%js%'))

```

react & render 18106

select \*:

TLE

Save 'JS' to increase speed

```

select *
from Posts p left join Comments c
on p.Id = c.PostId
where
(p.Title like '###key###' or p.Title like '###Method###') and
(
(p.Body like '###key###' and p.Body like '###Method###') or
(c.Text like '###key###' and c.Text like '###Method###')
)

```

Finally Works

```

select p.Title,p.CommentCount,p.Body,c.Text
from Posts p left join Comments c
on p.Id = c.PostId
where
(p.Title like '###key###' or p.Title like '###Method###') and
(
(p.Body like '###key###' and p.Body like '###Method###') or
(c.Text like '###key###' and c.Text like '###Method###')
)

```

and

```

select p.Title,p.CommentCount,p.Body,c.Text
from Posts p left join Comments c
on p.Id = c.PostId
where
(p.Title like '###key###' or p.Title like '###Method###') and
(
(p.Body like '###key###' and p.Body like '###Method###') or
(c.Text like '###key###' and c.Text like '###Method###')
)
and
(( p.Body like '%js%')
or
(c.Text like '%js%'))

```

Add Order by

```

select p.Id,p.Title,p.CommentCount,p.Body,c.Text
from Posts p left join Comments c
on p.Id = c.PostId
where
(p.Title like '###key###' or p.Title like '###Method###') and
(
(p.Body like '###key###' and p.Body like '###Method###') or
(c.Text like '###key###' and c.Text like '###Method###')
)
and
(( p.Body like '%js%')
or
(c.Text like '%js%'))
order by p.Id

```

18106 rows

(Key = react & Method = render)

Try to add TRIM

```

select p.Id,p.Title,p.CommentCount,p.Body,c.Text
from Posts p left join Comments c
on p.Id = c.PostId
where
(REPLACE(LTRIM(RTRIM(p.Title)),' ','') like '###key###' or REPLACE(LTRIM(RTRIM(p.Title)),' ','') like '###Method###') and
(
(REPLACE(LTRIM(RTRIM(p.Body)),' ','') like '###key###' and REPLACE(LTRIM(RTRIM(p.Body)),' ','') like '###Method###') or
(REPLACE(LTRIM(RTRIM(c.Text)),' ','') like '###key###' and REPLACE(LTRIM(RTRIM(c.Text)),' ','') like '###Method###')
)
and
(( REPLACE(LTRIM(RTRIM(p.Body)),' ','') like '%js%')
or
(REPLACE(LTRIM(RTRIM(c.Text)),' ','') like '%js%'))
order by p.Id

```

18205 rows returned in 256994 ms

(Key = react & Method = render)

Data Review:

Not all are related to ReactJs.

```

...public String execute() {
    ...countries = getCountries();
    ...return SUCCESS;
}

private Map<String, String> getCountries() {
}

```

ult - 171,469 hits

Line 9285: React.DOM.br(),

Line 9296: React.renderComponent(p, document.body);

Line 9305: "23067765", "Include component mount point contents in rendered component", "0", "<p>I have an application (built in Rails) that renders some markup in the back-en

Line 9309: <pre><code><!-- React\_component TestComponent do %>

Line 9316: <p>This call creates a Wrapper div with some <code>data=</code> attributes that react\_ujs then captures to instantiate React components. What I'm failing to do

Line 9316: <p>This call creates a wrapper div with some <code>data=</code> attributes that react\_ujs then captures to instantiate React components. What I'm failing to do

Line 9316: <p>This call creates a wrapper div with some <code>data=</code> attributes that react\_ujs then captures to instantiate React components. What I'm failing to do

Line 9318: <pre><code>React.createClass({

Line 9331: <p>Any ideas how this behavior can be achieved with ReactJS? I've received some pointers towards <code>this.props.children</code>, but this doesn't seem to yiel

Line 9334: It should actually react to a link-click, so it's </p>

Line 9362: It should actually react to a link-click, so it's </p>

Line 9389: It should actually react to a link-click, so it's </p>

Line 9416: It should actually react to a link-click, so it's </p>

Line 9442: "23337006", "Displaying a server side rendered reactJS component within a PHP application", "0", "<p>I have previously used reactJS on another project.

Line 9442: "23337006", "Displaying a server side rendered reactJS component within a PHP application", "0", "<p>I have previously used reactJS on another project.

Line 9445: <p>I am rebuilding an old project and I would like to replace some of the Javascript with ReactJS.</p>

Line 9451: <p>To achieve this my research implies that I should be using React's <code>React.renderComponentToString</code></p>

Line 9451: <p>To achieve this my research implies that I should be using React's <code>React.renderComponentToString</code></p>

Line 9455: <p>Pete Hunt mentions here (<a href="https://groups.google.com/forum/#!msg/reactjs/eUeepJFdyas/hmJywy09Z3wJ" rel="nofollow noreferrer">https://groups.google

Line 9455: <p>Pete Hunt mentions here (<a href="https://groups.google.com/forum/#!msg/reactjs/eUeepJFdyas/hmJywy09Z3wJ" rel="nofollow noreferrer">https://groups.google

Line 9477: <p>My interceptor does react on the 401 status and redirects to the appropriate route but at the same time it renders the previous route too. So I have a very u

Line 9546: <p>My interceptor does react on the 401 status and redirects to the appropriate route but at the same time it renders the previous route too. So I have a very u

Line 9615: <p>My interceptor does react on the 401 status and redirects to the appropriate route but at the same time it renders the previous route too. So I have a very u

Line 9684: <p>My interceptor does react on the 401 status and redirects to the appropriate route but at the same time it renders the previous route too. So I have a very u

Line 9753: <p>My interceptor does react on the 401 status and redirects to the appropriate route but at the same time it renders the previous route too. So I have a very u

Line 9822: <p>My interceptor does react on the 401 status and redirects to the appropriate route but at the same time it renders the previous route too. So I have a very u

text file length : 58,833,663 lines : 1,631, Ln : 2

## Differncr between TRIM

```

(select p.Id,p.Title,p.CommentCount,p.Body,c.Text
from Posts p left join Comments c
on p.Id = c.PostId
where
(REPLACE(LTRIM(RTRIM(p.Title)), ' ','') like '###key###' or REPLACE(LTRIM(RTRIM(p.Title)), ' ','') like '###Method###') and
(
(REPLACE(LTRIM(RTRIM(p.Body)), ' ','') like '###key###' and REPLACE(LTRIM(RTRIM(p.Body)), ' ','') like '###Method###') or
(REPLACE(LTRIM(RTRIM(c.Text)), ' ','') like '###key###' and REPLACE(LTRIM(RTRIM(c.Text)), ' ','') like '###Method###')
)
and
(( REPLACE(LTRIM(RTRIM(p.Body)), ' ','') like '%js%')
or
(REPLACE(LTRIM(RTRIM(c.Text)), ' ','') like '%js%'))
)
except
(
select p.Id,p.Title,p.CommentCount,p.Body,c.Text
from Posts p left join Comments c
on p.Id = c.PostId
where
(p.Title like '###key###' or p.Title like '###Method###') and
(
(p.Body like '###key###' and p.Body like '###Method###') or
(c.Text like '###key###' and c.Text like '###Method###')
)
and
(( p.Body like '%js%')
or
(c.Text like '%js%'))
)
)

```

TLE

Top 1000 still TLE...

```
select Top 10 * from Posts p
where p.Body like '###Key###'
or p.Body like '###Method###'
```