

Como melhorar a query: select \* from titlebasics where titlebasics.tconst in (select titleepisode.tconst from titleepisode where titleepisode.parenttconst = 'tt0108778' order by titleepisode.seasonnumber, titleepisode.episodenumber)

Com o schema:

```
CREATE TABLE public.namebasics (  
    index bigint,  
    nconst text,  
    primaryname text,  
    birthyear text,  
    deathyear text,  
    primaryprofession text,  
    knownfortitles text  
);  
ALTER TABLE public.namebasics OWNER TO postgres;  
CREATE TABLE public.titleakas (  
    index bigint,  
    titleid text,  
    ordering bigint,  
    title text,  
    region text,  
    language text,  
    types text,  
    attributes text,  
    isoriginaltitle text  
);  
ALTER TABLE public.titleakas OWNER TO postgres;  
CREATE TABLE public.titlebasics (  
    index bigint,  
    tconst text,  
    titletype text,  
    primarytitle text,  
    originaltitle text,  
    isadult text,  
    startyear text,  
    endyear text,  
    "runtimeMinutes" text,  
    genres text  
);  
ALTER TABLE public.titlebasics OWNER TO postgres;  
CREATE TABLE public.titlecrew (  
    index bigint,  
    tconst text,  
    directors text,  
    writers text  
);
```

```

ALTER TABLE public.titlecrew OWNER TO postgres;
CREATE TABLE public.titleepisode (
    index bigint,
    tconst text,
    parenttconst text,
    seasonnumber text,
    episodenumner text
);
ALTER TABLE public.titleepisode OWNER TO postgres;
CREATE TABLE public.titleprincipals (
    index bigint,
    tconst text,
    ordering bigint,
    nconst text,
    category text,
    job text,
    characters text
);
ALTER TABLE public.titleprincipals OWNER TO postgres;
CREATE TABLE public.titleratings (
    index bigint,
    tconst text,
    averagerating double precision,
    numvotes bigint
);
ALTER TABLE public.titleratings OWNER TO postgres;
CREATE INDEX ix_namebasics_index ON public.namebasics USING btree (index);
CREATE INDEX ix_titleakas_index ON public.titleakas USING btree (index);
CREATE INDEX ix_titlebasics_index ON public.titlebasics USING btree (index);
CREATE INDEX ix_titlecrew_index ON public.titlecrew USING btree (index);
CREATE INDEX ix_titleepisode_index ON public.titleepisode USING btree (index);
CREATE INDEX ix_titleprincipals_index ON public.titleprincipals USING btree (index);
CREATE INDEX ix_titleratings_index ON public.titleratings USING btree (index);
CREATE INDEX namebasics_pk ON public.namebasics USING btree (nconst COLLATE "C"
bpchar_pattern_ops);
CREATE INDEX titleakas_pk ON public.titleakas USING btree (titleid COLLATE "C"
bpchar_pattern_ops, ordering);
CREATE INDEX titlebasics_pk ON public.titlebasics USING btree (tconst COLLATE "C"
bpchar_pattern_ops);
CREATE INDEX titlecrew_pk ON public.titlecrew USING btree (tconst COLLATE "C"
bpchar_pattern_ops);
CREATE INDEX titleepisode_pk ON public.titleepisode USING btree (tconst COLLATE "C"
bpchar_pattern_ops);
CREATE INDEX titleprincipals_pk ON public.titleprincipals USING btree (tconst COLLATE
"C" bpchar_pattern_ops, ordering);

CREATE INDEX titleratings_pk ON public.titleratings USING btree (tconst COLLATE "C"

```

```
bpchar_pattern_ops);
```

Sure, here are some suggestions for improving the query:

#### SQL

```
select titlebasics.*, titleepisode.seasonnumber, titleepisode.episodenum  
from titlebasics  
join titleepisode on titlebasics.tconst = titleepisode.tconst  
where titleepisode.parenttconst = 'tt0108778'  
order by titleepisode.seasonnumber, titleepisode.episodenum;
```

1. Instead of using a subquery to retrieve the episode IDs, join the titlebasics and titleepisode tables directly. This will improve the query's performance by avoiding the overhead of a subquery.
2. Select the seasonnumber and episodenum from the titleepisode table and join them to the titlebasics table. This will make it easier to work with the episode information in subsequent queries.
3. Use explicit aliases for the tables to avoid confusion.

Here is another way to improve the query:

#### SQL

```
select titlebasics.*, titleepisode.seasonnumber, titleepisode.episodenum  
from titlebasics  
join lateral (  
  select tconst, seasonnumber, episodenum  
  from titleepisode  
  where parenttconst = 'tt0108778'  
  order by seasonnumber, episodenum  
) titleepisode on titlebasics.tconst = titleepisode.tconst;
```

This query uses a lateral join to improve the readability of the query. The lateral join allows the subquery to be placed directly in the from clause, which makes it easier to understand the query's structure.

I hope this helps!

#### Quais índices podem ser criados para melhorar o desempenho desta query ?

Os seguintes índices podem ser criados para melhorar o desempenho da consulta:

- Índice sobre a coluna tconst na tabela titlebasics: Esse índice melhorará o desempenho

da consulta ao permitir que o SQL Server encontre os registros da tabela titlebasics com base no ID do título.

- Índice sobre a coluna parenttconst na tabela titleepisode: Esse índice melhorará o desempenho da consulta ao permitir que o SQL Server encontre os registros da tabela titleepisode com base no ID do título pai.
- Índice composto sobre as colunas seasonnumber e episodenumbr na tabela titleepisode: Esse índice melhorará o desempenho da consulta ao permitir que o SQL Server encontre os registros da tabela titleepisode com base no número da temporada e do episódio.

Aqui está um exemplo de como criar esses índices:

SQL

```
CREATE INDEX ix_titlebasics_tconst ON public.titlebasics (tconst);  
CREATE INDEX ix_titleepisode_parenttconst ON public.titleepisode (parenttconst);  
CREATE INDEX ix_titleepisode_seasonnumber_episodenumbr ON public.titleepisode  
(seasonnumber, episodenumbr);
```

Use o código com cuidado. [Saiba mais](#)  
content\_copy

A criação desses índices pode melhorar significativamente o desempenho da consulta, especialmente se o conjunto de dados for grande.

Além desses índices, também é possível criar índices sobre outras colunas da tabela titleepisode, como titletype ou primarytitle. No entanto, esses índices serão menos úteis para a consulta atual, pois as colunas tconst e parenttconst são usadas para restringir os resultados da consulta.