



LABORATÓRIO DE BANCO DE DADOS

IBD100

Resolvendo questões em AR + SQL usando o pgadmin 4

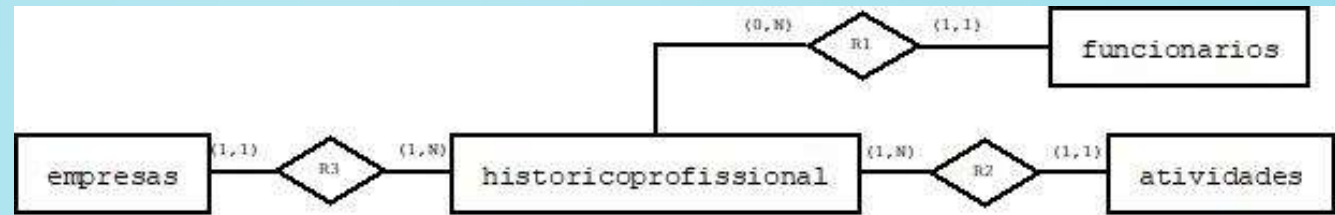
- Proposta e resolução de perguntas com uso do procedimento de divisão e encadeamento de comandos SQL para apresentar respostas com apenas um comando SQL

RESOLVENDO QUESTÕES AR+SQL

- Com base no Modelo de Dados da Base implementadas em PostgreSQL, apresentam-se as perguntas:
- Quais os nomes dos funcionários que exerceram todas as atividades registradas em históricos profissionais?
- Quais são as oficinas da cidade de "Morrinhos" que fazem serviços de "Ajuste na Suspensão"?

RESOLVENDO QUESTÕES AR+SQL

- Quais os nomes dos funcionários que exerceram todas as atividades registradas em históricos profissionais?



- `-- Preparação`
- `-- F <- funcionarios[[pkfuncionario->pkf]]`
- `-- PF <- historicoprofissional[[fkatividade->pka,fkfuncionario->pkf]]`
- `-- P <- Atividades[[pkatividade->pka]]`
- `-- Executando a divisão`
- `-- S <- P X F`
- `-- T <- PF - S`
- `-- K <- T[[pkf]]`
- `-- W <- F-K`
- `-- Completando a resposta`
- `-- G <- W[[w.pkf=funcionarios.pkfuncionario]]funcionarios`
- `-- G[[concatene(txprenomes, ' ',txnomefuncionario)->NomeCompleto]]`

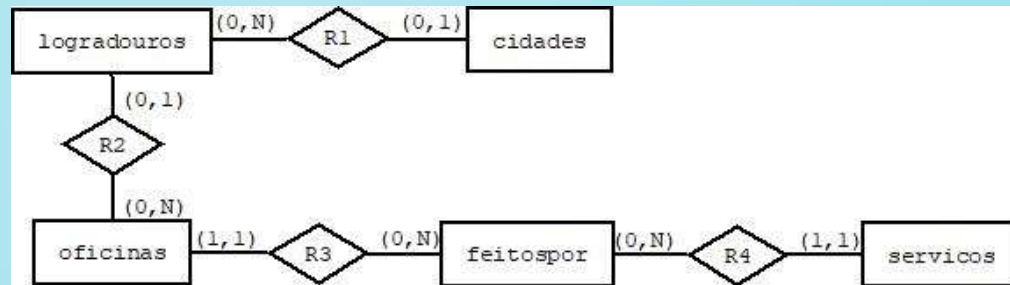
RESOLVENDO QUESTÕES AR+SQL

- Em SQL temos

- -- Preparação
- -- F <- funcionarios[[pkfuncionario->pkf]]
- `create temporary table F as (select pkfuncionario as pkf from funcionarios);`
- -- PF <- historicoprofissional[[fkatividade->pka,fkfuncionario->pkf]]
- `create temporary table PF as (select fkatividade as pka, fkfuncionario as pkf
from historicoprofissional);`
- -- P <- Atividades[[pkatividade->pka]]
- `create temporary table P as (select pkatividade as pka from atividades);`
- -- Executando a divisão
- -- S <- P X F
- `create temporary table S as (select * from P cross join F);`
- -- T <- S - PF
- `create temporary table T as (select * from S except (select * from PF));`
- -- K <- T[[pkf]]
- `create temporary table K as (select pkf from T);`
- -- W <- F-K
- `create temporary table W as (select * from F except (select * from k));`
- -- Completando a resposta
- -- G <- W[[W.pkf=funcionarios.pkfuncionario]]funcionarios
- `create table G as (select * from W inner join funcionarios
on W.pkf=funcionarios.pkfuncionario);`
- -- G[[concatene(txprenomes, ' ', txnomefuncionario)->NomeCompleto]]
- `select concat(txprenomes, ' ', txsobrenome) as NomeCompleto from G;`

RESOLVENDO QUESTÕES AR+SQL

- Quais são as oficinas da cidade de "Morrinhos" que fazem serviços de "Ajuste na Suspensão"?
- O Modelo de Dados é assim:



- A AR fica assim:
- $A \leftarrow cidades[[txnomecidade='Morrinhos']]$
- $B \leftarrow A[[A.pkcidade=L.fkcidade]]logradouros \rightarrow L$
- $C \leftarrow B[[B.pklogradouro=O.fklogradouro]]oficinas \rightarrow O$
- $D \leftarrow C[[C.pkoficina=F.fkoficina]]feitospor \rightarrow F$
- $E \leftarrow D[[D.fkservico=S.pkservico]]servicos \rightarrow S$
- $G \leftarrow E[[txnomecurto='Ajuste na Suspensão']]$
- $G[['txrazaosocial']]$

- Em SQL...

- A ← cidades[[txnomecidade='Morrinhos']]
- create temporary table A as (select * from cidades
where txnomecidade='Morrinhos');
- B ← A[[A.pkcidade=L.fkcidade]]logradouros → L
- create temporary table B as (select * from A inner join logradouros as L
on A.pkcidade=L.fkcidade);
- C ← B[[B.pklogradouro=O.fklogradouro]]oficinas → O
- create temporary table C as (select * from B inner join oficinas as O
on B.pklogradouro=O.fklogradouro);
- D ← C[[C.pkoficina=F.fkoficina]]feitospor → F
- create temporary table D as (select * from C inner join feitospor as F
on C.pkoficina=F.fkoficina);
- E ← D[[D.fkservico=S.pkservico]]serviços → S
- create temporary table E as (select * from D inner join servicos as S
on D.fkservico=S.pkservico);
- G ← E[[txnomecurto='Ajuste na Suspensão']]
- create temporary table G as (select * from E
where txnomecurto='Ajuste na Suspensão');
- G[['txrazaosocial']]
- select txrazaosocial from G;

- Porém (Ah! Porém...)

- A estrutura montada nos comandos seguintes apresenta 2x o campo dtfundação (em cidades e em oficinas). O PG apresenta um erro de execução... Então (eliminando a repetição do campo).
- `A←cidades[[txnomecidade='Morrinhos']]`
- `create temporary table A as (select * from cidades
where txnomecidade='Morrinhos');`
- `A1 ← A[[pkcidade]]`
- `create temporary table A1 as (select pkcidade from A);`
- `B←A1[[A1.pkcidade=L.fkcidade]]logradouros→L`
- `create temporary table B as (select * from A1 inner join logradouros as L
on A1.pkcidade=L.fkcidade);`
- `C←B[[B.pklogradouro=O.fklogradouro]]oficinas→O`
- `create temporary table C as (select * from B inner join oficinas as O
on B.pklogradouro=O.fklogradouro);`
- `D←C[[C.pkoficina=F.fkoficina]]feitospor→F`
- `create temporary table D as (select * from C inner join feitospor as F
on C.pkoficina=F.fkoficina);`
- `E←D[[D.fkservico=S.pkservico]]serviços→S`
- `create temporary table E as (select * from D inner join servicos as S
on D.fkservico=S.pkservico);`
- `G←E[[txnomecurto='Ajuste na Suspensão']]`
- `create temporary table G as (select * from E
where txnomecurto='Ajuste na Suspensão');`
- `G[['txrazaosocial']]`
- `select txrazaosocial from G;`

RESOLVENDO QUESTÕES AR+SQL

- Agora podemos escrever estes comandos em apenas um comando SQL (não criando tabelas temporárias).

- Saíndo da seleção de cidades com txnomecidade='Morrinhos' faz-se a junção com logradouros
- ```
select * from cidades inner join logradouros as L
 where txnomecidade='Morrinhos');
```
- Sem criar tabelas temporárias, faz a junção deste conjunto com oficinasA1 ← A[[pkcidade]]
- ```
select * from cidades as C inner join logradouros as L on C.pkcidade=L.fkcidade
      inner join oficinas as O on L.pklogradouro=O.fklogradouro
      where txnomecidade='Morrinhos');
```
- Agora faze-se a junção com feitospor
- ```
select * from cidades as C inner join logradouros as L on C.pkcidade=L.fkcidade
 inner join oficinas as O on L.pklogradouro=O.fklogradouro
 inner join feitospor as F on O.pkoficina=F.fkoficina)
 inner join servicos as S on F.fkservico=S.pkservico)
 where txnomecidade='Morrinhos');
```
- Agora simplesmente projeta-se o campo com os nomes das oficinas (txrazaosocial)
- ```
select txrazaosocial
      from cidades as C inner join logradouros as L on C.pkcidade=L.fkcidade
      inner join oficinas as O on L.pklogradouro=O.fklogradouro
      inner join feitospor as F on O.pkoficina=F.fkoficina)
      inner join servicos as S on F.fkservico=S.pkservico)
      where txnomecidade='Morrinhos');
```



LABORATÓRIO DE BANCO DE DADOS

IBD100