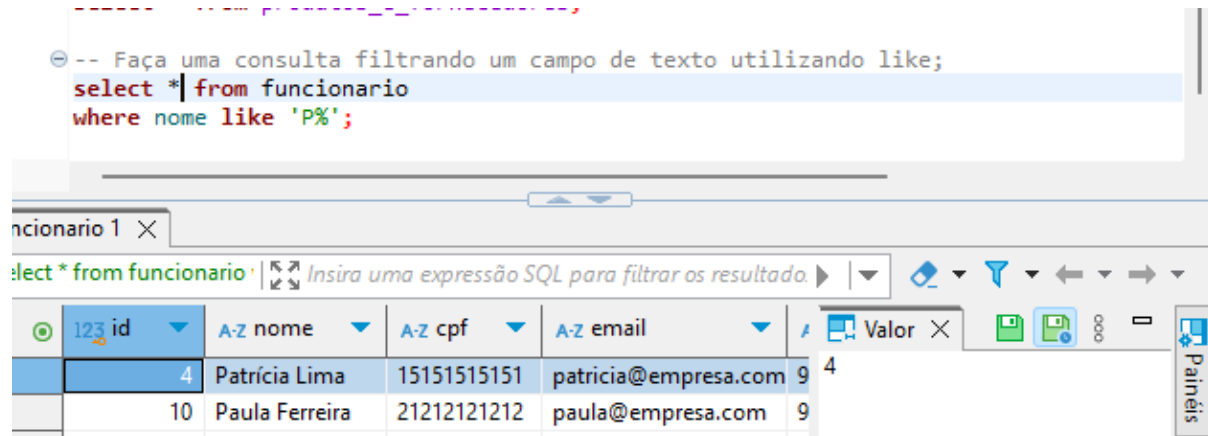


Processo documentado – aula do dia 28/08

Aluna: Giovanna Felisardo

Turma: 2 Info B

Faça uma consulta filtrando um campo de texto utilizando like;



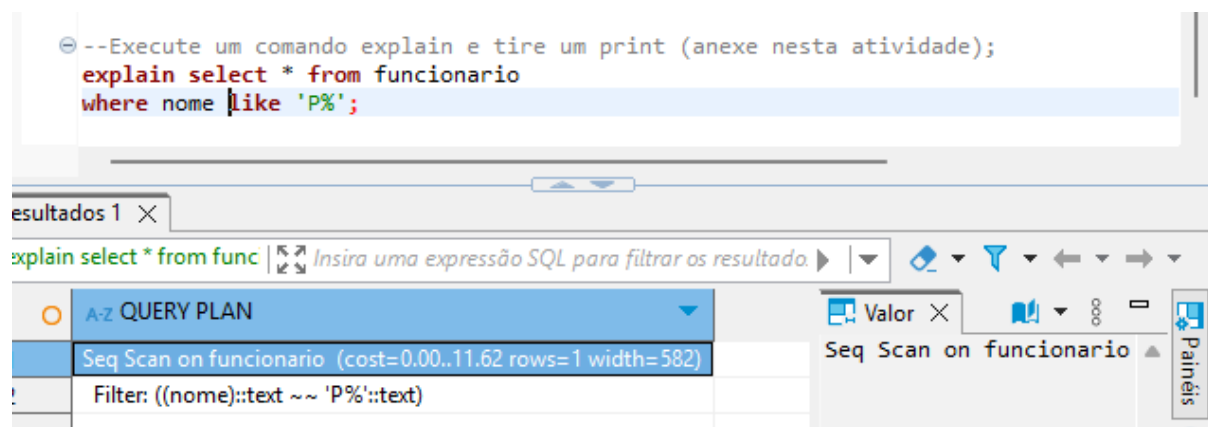
The screenshot shows a database interface with a SQL editor at the top and a results pane below. The SQL editor contains the following query:

```
-- Faça uma consulta filtrando um campo de texto utilizando like;  
select * from funcionario  
where nome like 'P%';
```

The results pane shows a table with the following data:

id	nome	cpf	email	Valor
4	Patrícia Lima	15151515151	patricia@empresa.com	9
10	Paula Ferreira	21212121212	paula@empresa.com	9

Execute um comando explain e tire um print (anexe nesta atividade);



The screenshot shows a database interface with a SQL editor at the top and a results pane below. The SQL editor contains the following query:

```
--Execute um comando explain e tire um print (anexe nesta atividade);  
explain select * from funcionario  
where nome like 'P%';
```

The results pane shows the query plan for the above query:

A-Z QUERY PLAN	Valor
Seq Scan on funcionario (cost=0.00..11.62 rows=1 width=582)	Seq Scan on funcionario
Filter: ((nome)::text ~~ 'P%':text)	

Crie um index para a coluna que utilizou no filtro acima;

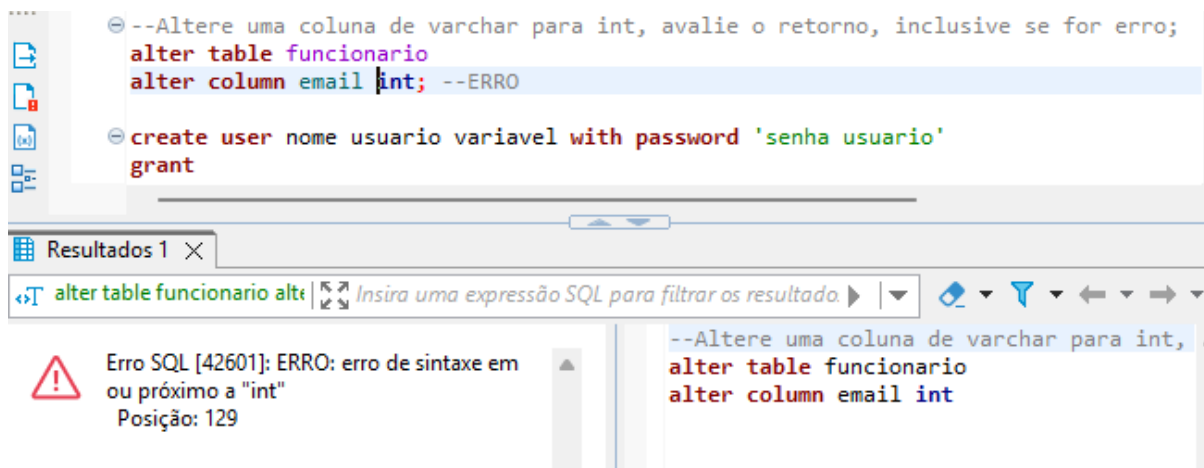
<pre>--Crie um index para a coluna que utilizou no filtro acima; create index idx_nome_fun on funcionario(nome);</pre>	
Estatísticas 1 X	
Name	Value
Updated Rows	0
Query	--Crie um index para a coluna que utilizou no filtro acima; create index idx_nome_fun on funcionario(nome)
Start time	Thu Aug 28 08:55:18 BRT 2025
Finish time	Thu Aug 28 08:55:18 BRT 2025

Refaça a consulta e execute o explain novamente. Tire um novo print (anexe neste atividade) e compare com o anterior. Aponte as diferenças.

<pre>--Refaça a consulta e execute o explain novamente. Tire um novo print --(anexe neste atividade) e compare com o anterior. Aponte as diferenças. explain select * from funcionario where nome like 'P%';</pre>	
Resultados 1 X	
<pre>explain select * from func</pre> <i>Insira uma expressão SQL para filtrar os resultados.</i>	
Grade	A-Z QUERY PLAN
1	Seq Scan on funcionario (cost=0.00..1.12 rows=1 width=582)
2	Filter: ((nome)::text ~~ 'P% '::text)
Texto	

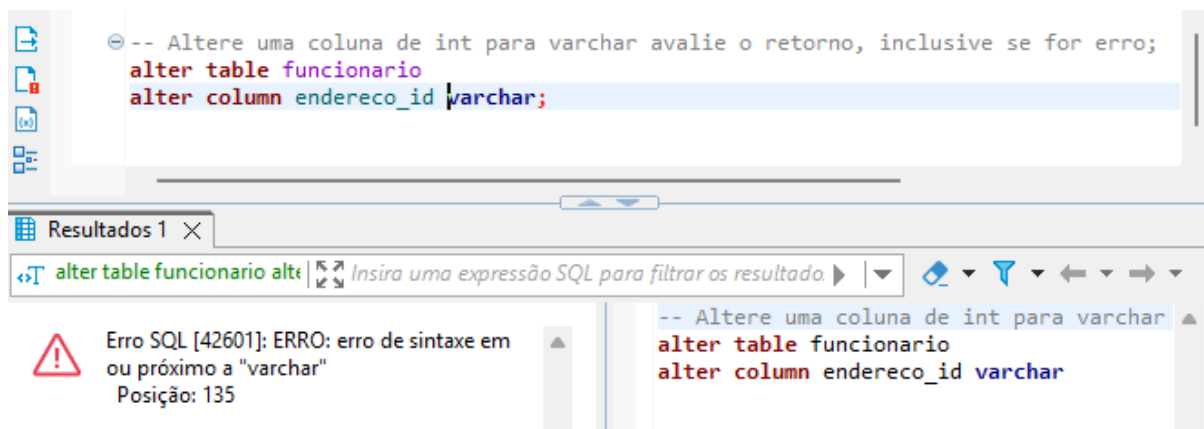
Resposta: As duas saídas estão iguais. Não tem diferença.

Altere uma coluna de varchar para int, avalie o retorno, inclusive se for erro;



Deu erro porque a variável 'email' não pode ter seu valor como int, por isso esse comando não pode ser executado.

Altere uma coluna de int para varchar avalie o retorno, inclusive se for erro;



Deu erro porque a variável 'endereco_id' não pode ter seu valor como varchar, por isso esse comando não pode ser executado.

Crie um usuário com seu nome e dê todas as permissões de acesso para todas as tabelas;

```
--Crie um usuário com seu nome e dê todas as permissões de acesso para  
--todas as tabelas;  
create user eu with password '1234';  
grant all privileges on database db_revenda_giovanna to eu;
```

Crie um usuário para seu colega apenas com permissão de select em uma das tabelas;

```
create user amiguinho with password '5678';  
grant select on cliente to amiguinho;
```

Refaça todos os itens no usuário que criou para seu colega, registre tudo que ocorreu (erros e acertos).

```
-- Faça uma consulta filtrando um campo de texto utilizando like;
select * from funcionario
where nome like 'P%';
```

Resultados 1 X

`select * from funcionario` | *Insira uma expressão SQL para filtrar os resultados.*

Erro SQL [42501]: ERRO: permissão negada para tabela funcionario

```
-- Faça uma consulta filtrando um campo de texto utilizando like;
select * from funcionario
where nome like 'P%';
```

```
--Execute um comando explain e tire um print (anexe nesta atividade);
explain select * from funcionario
where nome like 'P%';
```

Resultados 1 X

`explain select * from func` | *Insira uma expressão SQL para filtrar os resultados.*

Erro SQL [42501]: ERRO: permissão negada para tabela funcionario

```
--Execute um comando explain e tire um print (anexe nesta atividade);
explain select * from funcionario
where nome like 'P%';
```

```
--Crie um index para a coluna que utilizou no filtro acima;
create index idx_nome_fun
on funcionario(nome);

--Refaça a consulta e execute o explain novamente. Tire um novo print
--(anexe nesta atividade) e compare com o anterior. Aponte as diferenças.
```

Resultados 1 X

`create index idx_nome_fui` | *Insira uma expressão SQL para filtrar os resultados.*

Erro SQL [42501]: ERRO: é necessário ser o dono da tabela funcionario

```
--Crie um index para a coluna que utilizou no filtro acima;
create index idx_nome_fun
on funcionario(nome);
```

```
--Altere uma coluna de varchar para int, avalie o retorno, inclusive se for erro;
alter table funcionario
alter column email int; --ERRO
```

Resultados 1 X

alter table funcionario alt; Insira uma expressão SQL para filtrar os resultados.

Erro SQL [42601]: ERRO: erro de sintaxe em ou próximo a "int"
Posição: 129

```
--Altere uma coluna de varchar para int,
alter table funcionario
alter column email int
```

```
-- Altere uma coluna de int para varchar avalie o retorno, inclusive se for erro;
alter table funcionario
alter column endereco_id varchar; --ERRO
```

Resultados 1 X

alter table funcionario alt; Insira uma expressão SQL para filtrar os resultados.

Erro SQL [42601]: ERRO: erro de sintaxe em ou próximo a "varchar"
Posição: 135

Posição do erro: line: 195 pos: 134

```
-- Altere uma coluna de int para varchar
alter table funcionario
alter column endereco_id varchar
```

Com os itens que não deram erro da primeira vez a ser executado apareceu uma mensagem falando que o usuário 'amiguinho' não tinha permissão para a função. Já nos itens que já estavam dando erro, apareceu apenas o erro já comentado aqui.

De volta no seu usuário, crie 12 consultas, sendo 3 consultas semelhantes (somente com alteração do tipo de join: inner, left e right). Ou seja, são 4 consultas diferentes, sendo que cada consulta terá 3 versões, uma com cada tipo de join.

```

select nome, endereco_id, cidade
from funcionario
inner join endereco
on funcionario.endereco_id = endereco.id;

select nome, endereco_id, cidade
from funcionario
left join endereco
on funcionario.endereco_id = endereco.id;

select nome, endereco_id, cidade
from funcionario
right join endereco
on funcionario.endereco_id = endereco.id;
-----

select nome, endereco_id, bairro
from cliente
inner join endereco
on cliente.endereco_id = endereco.id

select nome, endereco_id, bairro
from cliente
left join endereco
on cliente.endereco_id = endereco.id

select nome, endereco_id, bairro
from cliente
right join endereco
on cliente.endereco_id = endereco.id
-----

select nome_fornecedor, endereco_id, cidade
from fornecedor
inner join endereco
on fornecedor.endereco_id = endereco.id;

select nome_fornecedor, endereco_id, cidade
from fornecedor
left join endereco
on fornecedor.endereco_id = endereco.id;

select nome_fornecedor, endereco_id, cidade
from fornecedor
right join endereco
on fornecedor.endereco_id = endereco.id;
-----

select nome_fornecedor, valor_prod_fornecedor, valor_produto, fornecedor_id
from fornecedor
inner join produtos
on fornecedor.id = produtos.fornecedor_id;

select nome_fornecedor, valor_prod_fornecedor, valor_produto, fornecedor_id
from fornecedor
left join produtos
on fornecedor.id = produtos.fornecedor_id;

select nome_fornecedor, valor_prod_fornecedor, valor_produto, fornecedor_id
from fornecedor
right join produtos
on fornecedor.id = produtos.fornecedor_id;

```

Atualize vários registros com colunas NULL;

```

--Atualize vários registros com colunas NULL;
alter table funcionario
alter column nome drop not null;

alter table endereco
alter column cidade drop not null;

alter table endereco
alter column bairro drop not null;

alter table cliente
alter column nome drop not null;

alter table fornecedor
alter column valor_prod_fornecedor drop not null;

```

Execute as consultas com Join novamente, avalie os resultados.

```

--Execute as consultas com Join novamente, avalie os resultados.
select nome, endereco_id, cidade
from funcionario
inner join endereco
on funcionario.endereco_id = endereco.id;

select nome, endereco_id, cidade

```

funcionario(+) 1 X

select nome, endereco_id, Insira uma expressão SQL para filtrar os resultados

	A-Z nome	123 endereco_id	A-Z cidade
1		21	Santos
2	Fernanda Souza	22	Petrópolis
3	Patrícia Lima	24	Gramado
4	Gustavo Ramos	25	Porto Seguro
5	Mariana Dias	28	Sobral
6	Eduardo Martins	29	Blumenau
7	Paula Ferreira	30	Rio Verde
8	Ricardo Melo	23	
9		26	
10	Thiago Costa	27	

```
select nome, endereco_id, cidade
from funcionario
left join endereco
on funcionario.endereco_id = endereco.id;
```

```
select nome, endereco_id, cidade
from funcionario
```

funcionario(+) 1 ✕


 select nome, endereco_id,  *Insira uma expressão SQL para filtrar os resultados.*

	A-Z nome	123 endereco_id	A-Z cidade
1		21	Santos
2	Fernanda Souza	22	Petrópolis
3	Patrícia Lima	24	Gramado
4	Gustavo Ramos	25	Porto Seguro
5	Mariana Dias	28	Sobral
6	Eduardo Martins	29	Blumenau
7	Paula Ferreira	30	Rio Verde
8	Ricardo Melo	23	
9		26	
10	Thiago Costa	27	

```
select nome, endereco_id, cidade
from funcionario
right join endereco
on funcionario.endereco_id = endereco.id;
```

funcionario(+) 1 ✕

select nome, endereco_id, Insira uma expressão SQL para filtrar os resultados.

Grade		A-Z nome	123 endereco_id	A-Z cidade
<div> <div>Texto</div> <div>registro</div> </div>	1	[NULL]	[NULL]	Rio de Janeiro
	2	[NULL]	[NULL]	Belo Horizonte
	3	[NULL]	[NULL]	Curitiba
	4	[NULL]	[NULL]	Recife
	5	[NULL]	[NULL]	Fortaleza
	6	[NULL]	[NULL]	Florianópolis
	7	[NULL]	[NULL]	Campinas
	8	[NULL]	[NULL]	Niterói
	9	[NULL]	[NULL]	Uberlândia
	10	[NULL]	[NULL]	Feira de Santana
	11	[NULL]	[NULL]	Londrina
	12	[NULL]	[NULL]	Olinda
	13	[NULL]	[NULL]	Joinville


```

- select nome, endereco_id, bairro
  from cliente
  inner join endereco
    on cliente.endereco_id = endereco.id

- select nome, endereco_id, bairro

```


cliente(+) 1 ✕

 `select nome, endereco_id,` *Insira uma expressão SQL para filtrar os resultados.*

	A-Z nome	123 endereco_id	A-Z bairro
1	Ana Souza	1	Centro
2	João Lima	2	Copacabana
3	Pedro Santos	4	
4	Lucas Rocha	5	Barra
5	Julia Reis	6	Centro
6	Camila Nunes	8	Aldeota
7	Felipe Araújo	9	Trindade
8	Beatriz Gomes	10	
9		3	Savassi
10		7	Boa Viagem

```
select nome, endereco_id, bairro
from cliente
left join endereco
on cliente.endereco_id = endereco.id
```

cliente(+) 1 ✕

 `select nome, endereco_id,`  *Insira uma expressão SQL para filtrar os resultados.*

	A-Z nome	123 endereco_id	A-Z bairro
1	Ana Souza	1	Centro
2	João Lima	2	Copacabana
3	Pedro Santos	4	
4	Lucas Rocha	5	Barra
5	Julia Reis	6	Centro
6	Camila Nunes	8	Aldeota
7	Felipe Araújo	9	Trindade
8	Beatriz Gomes	10	
9		3	Savassi
10		7	Boa Viagem

```
select nome, endereco_id, bairro
from cliente
right join endereco
on cliente.endereco_id = endereco.id
```

cliente(+) 1 X

select nome, endereco_id, Insira uma expressão SQL para filtrar os resultado.

Grade	A-Z nome	123 endereco_id	A-Z bairro
4	Lucas Rocha	5	Barra
5	Julia Reis	6	Centro
6	Camila Nunes	8	Aldeota
7	Felipe Araújo	9	Trindade
8	Beatriz Gomes	10	
9		3	Savassi
10		7	Boa Viagem
11	[NULL]	[NULL]	
12	[NULL]	[NULL]	Arraial d'Ajuda
13	[NULL]	[NULL]	Zona 7
14	[NULL]	[NULL]	Centro
15	[NULL]	[NULL]	Taquaral
16	[NULL]	[NULL]	Carmo

```
select nome_fornecedor, endereco_id, cidade
from fornecedor
inner join endereco
on fornecedor.endereco_id = endereco.id;
```

```
select nome_fornecedor, endereco_id, cidade
```

fornecedor(+) 1 X

select nome_fornecedor, e, Insira uma expressão SQL para filtrar os resultado.

A-Z nome_fornecedor	123 endereco_id	A-Z cidade
Fornecedor A	11	Campinas
Fornecedor C	13	Uberlândia
Fornecedor D	14	
Fornecedor F	16	Londrina
Fornecedor G	17	Olinda
Fornecedor H	18	Juazeiro do Norte
Fornecedor I	19	Joinville
Fornecedor J	20	Anápolis
Fornecedor B	12	Niterói
Fornecedor E	15	Feira de Santana

SQL query editor showing a left join operation between the `fornecedor` and `endereco` tables.

```
select nome_fornecedor, endereco_id, cidade
from fornecedor
left join endereco
on fornecedor.endereco_id = endereco.id;
```

Query results (Table 1):

Grade	A-Z nome_fornecedor	123 endereco_id	A-Z cidade
1	Fornecedor A	11	Campinas
2	Fornecedor C	13	Uberlândia
3	Fornecedor D	14	
4	Fornecedor F	16	Londrina
5	Fornecedor G	17	Olinda
6	Fornecedor H	18	Juazeiro do Norte
7	Fornecedor I	19	Joinville
8	Fornecedor J	20	Anápolis
9	Fornecedor B	12	Niterói
10	Fornecedor E	15	Feira de Santana

SQL query editor showing a right join operation between the `fornecedor` and `endereco` tables.

```
select nome_fornecedor, endereco_id, cidade
from fornecedor
right join endereco
on fornecedor.endereco_id = endereco.id;
```

Query results (Table 2):

Grade	A-Z nome_fornecedor	123 endereco_id	A-Z cidade
7	Fornecedor I	19	Joinville
8	Fornecedor J	20	Anápolis
9	Fornecedor B	12	Niterói
10	Fornecedor E	15	Feira de Santana
11	[NULL]	[NULL]	Porto Seguro
12	[NULL]	[NULL]	
13	[NULL]	[NULL]	
14	[NULL]	[NULL]	Goiânia
15	[NULL]	[NULL]	Rio de Janeiro
16	[NULL]	[NULL]	Santos
17	[NULL]	[NULL]	
18	[NULL]	[NULL]	Gramado
19	[NULL]	[NULL]	Fortaleza


```

❶ select nome_fornecedor, valor_prod_fornecedor, valor_produto, fornecedor_id
   from fornecedor
  inner join produtos
 on fornecedor.id = produtos.fornecedor_id;

```

```
➤ select nome fornecedor, valor prod fornecedor, valor produto, fornecedor id
```


fornecedor(+) 1 ✕

select nome_fornecedor, v  *Insira uma expressão SQL para filtrar os resultados.*

Grade	A-Z nome_fornecedor	123 valor_prod_fornecedor	123 valor_produto	123 fornecedor_id
1	Fornecedor A	50	120	
2	Fornecedor B	[NULL]	350	
3	Fornecedor C	20	200	
4	Fornecedor D	70	180	
5	Fornecedor E	[NULL]	220	
6	Fornecedor F	60	150	
7	Fornecedor G	40	90	
8	Fornecedor H	15	250	
9	Fornecedor I	120	280	
10	Fornecedor J	35	100	

```
select nome_fornecedor, valor_prod_fornecedor, valor_produto, fornecedor_id
from fornecedor
left join produtos
on fornecedor.id = produtos.fornecedor_id;
```

fornecedor(+) 1 ✕

 `select nome_fornecedor, v`  *Insira uma expressão SQL para filtrar os resultados.* 

Grade	A-Z nome_fornecedor	123 valor_prod_fornecedor	123 valor_produto	123 fornecedor_id
1	Fornecedor A	50	120	
2	Fornecedor B	[NULL]	350	
3	Fornecedor C	20	200	
4	Fornecedor D	70	180	
5	Fornecedor E	[NULL]	220	
6	Fornecedor F	60	150	
7	Fornecedor G	40	90	
8	Fornecedor H	15	250	
9	Fornecedor I	120	280	
10	Fornecedor J	35	100	

select nome_fornecedor, valor_prod_fornecedor, valor_produto, fornecedor_id
 from fornecedor
 right join produtos
 on fornecedor.id = produtos.fornecedor_id;

fornecedor(+) 1 X

select nome_fornecedor, v | Insira uma expressão SQL para filtrar os resultados.

	A-Z nome_fornecedor	123 valor_prod_fornecedor	123 valor_produto	123 fornecedor_id
1	Fornecedor A	50	120	
2	Fornecedor B	[NULL]	350	
3	Fornecedor C	20	200	
4	Fornecedor D	70	180	
5	Fornecedor E	[NULL]	220	
6	Fornecedor F	60	150	
7	Fornecedor G	40	90	
8	Fornecedor H	15	250	
9	Fornecedor I	120	280	
10	Fornecedor J	35	100	

Resultado: Na maioria dos selects com inner join e left join, mesmo após deixar alguns campos nulos, eles foram iguais. Já o right join ficou diferente na maioria dos casos.