

GuiaPG-5: Criando PKs e FKs

Criando Chaves

Uma das funcionalidades do SGBD PostgreSQL é criar e administrar campos definidos como chaves primárias, chaves estrangeiras e chaves candidatas. Estes conceitos são definidos no PG como regras de limitação do comportamento do dado. As regras de limitação dos dados são caracterizadas como “CONSTRAINTS”. No PGAdmin 4 é possível então definir as constraints de chave primárias, de chaves estrangeiras e chaves candidatas. É este o objetivo deste Guia.

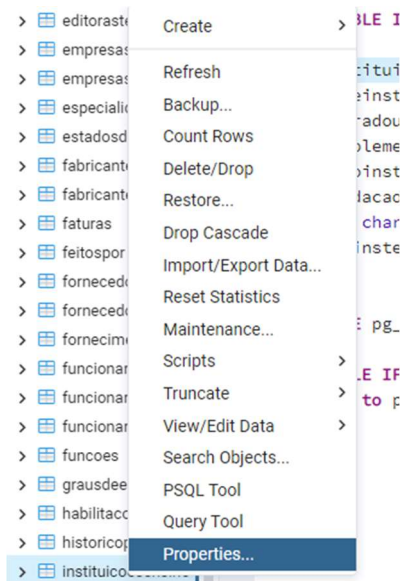
Chaves Primárias

As PKs (Primaries Keys) têm um modo muito semelhante de serem definidas, tais como as FKs (Foreign Keys) e UKs (Unique Keys). Pré-Requisito: SE a tabela já conter tuplas, então o(s) campo(s) que forma(m) a PK NÃO PODEM apresentar dados repetidos.

Seguindo o roteiro:

Acessando a estrutura da tabela

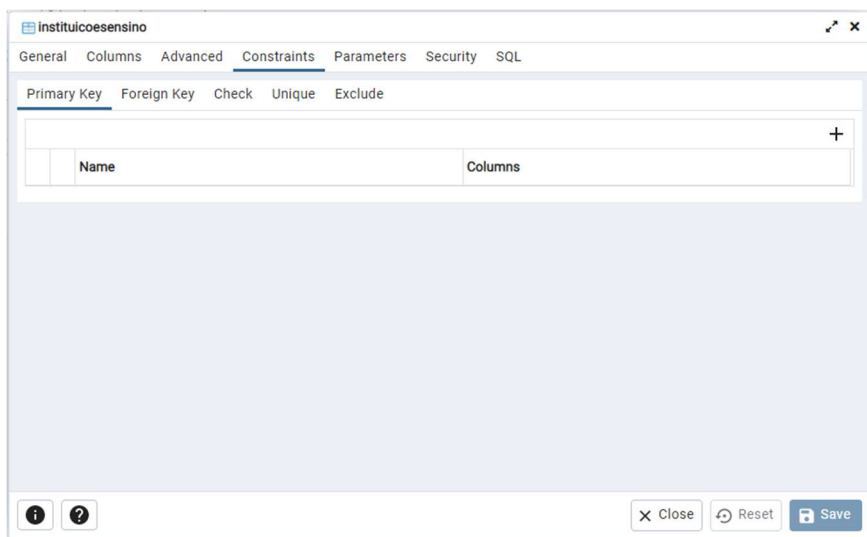
Escolhi a tabela instituicoesensino:



Ao clicar o botão secundário do mouse com o cursor sobre o nome da tabela vemos o menu popup de contexto.

Basta então escolher a alternativa Properties...

A tela que se apresenta mostra as abas superiores e os botões de finalização de ação na parte inferior.



Acessando a Aba Constraints

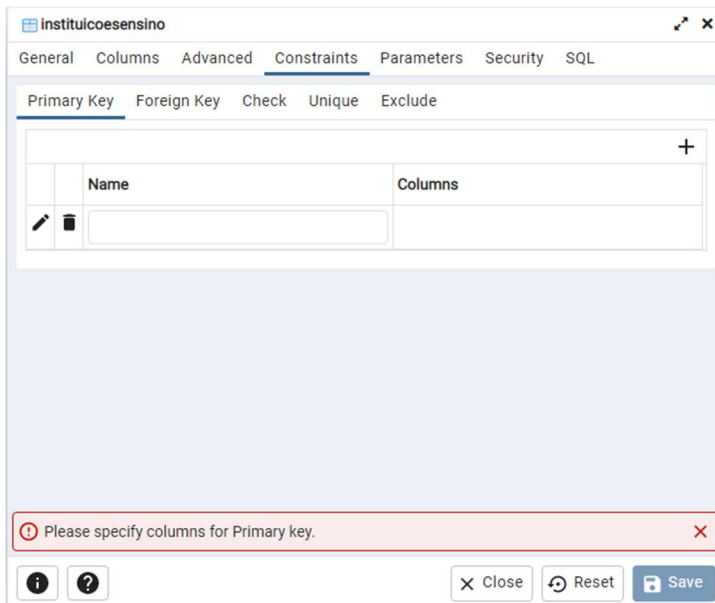
Clique na Aba constraint e a seguir na sub-aba primary key.

No lado direito da tela vê-se um sinal de “+”. Clique neste sinal para iniciar o preenchimento da tela com os dados da PK.

A tela mostra uma mensagem de erro: “Please especify columns for primary key”. Calma.

A PK deve ser nomeada e referenciada a um campo da tabela.

Entrando com o nome da PK

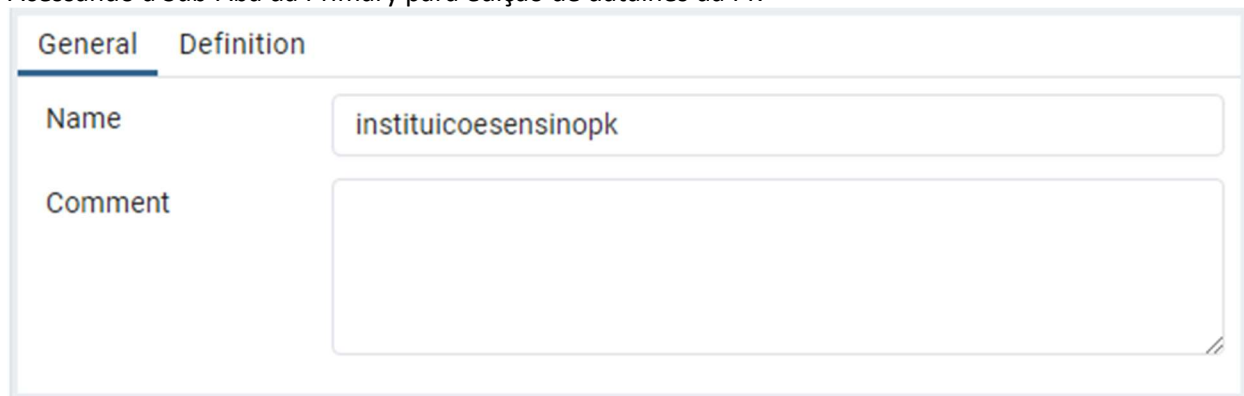


Em qualquer base de dados do PostgreSQL, o nome de uma chave primária não pode se repetir. Então é uma sugestão fortemente recomendada que os nomes de chaves sejam atrelados ao nome da tabela.

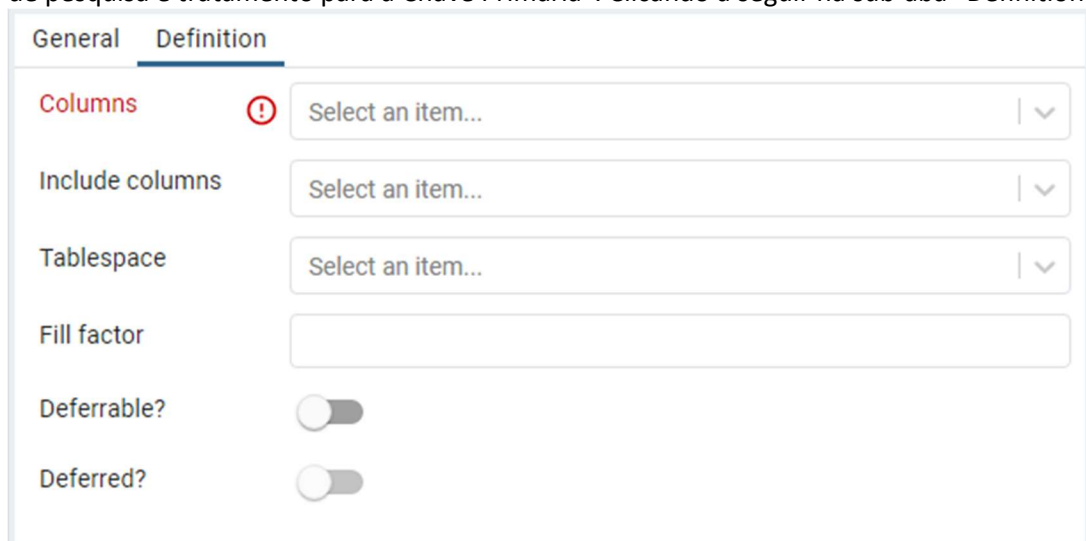
Desta forma nomeamos a PK com o nome da tabela seguido de 'pk'. Assim a PK que estamos criando neste exemplo tem nome: `instituoensinopk`.

Depois de preencher o nome, deve-se clicar no ícone de um lápis (à esquerda do campo com o nome da PK) para acessar a tela que permite informar os dados de formação da PK.

Acessando a Sub-Aba da Primary para edição de detalhes da PK



No campo COMMENT você pode escrever algum texto para caracterizar a PK. NOTE BEM: o nome da estrutura que você está criando é a referência que o PostgreSQL usará nos algoritmos de tratamento e acesso aos dados da tabela. Os programadores e analista NÃO precisam fazer a referência em comandos SQL para usar estas estruturas. Por este motivo, tem analista que não escreve nada neste campo. Eu aconselho escrever “Estrutura de pesquisa e tratamento para a Chave Primária”. Clicando a seguir na sub-aba “Definition” vemos:



Clicando no campo ‘columns’ veremos uma lista de campos da tabela para que se escolha o campo que forma a PK. Depois de escolhido o primeiro, clicando imediatamente à direita, vê-se a lista e a possibilidade de escolher outro campo (em caso de PK Composta).

General Definition

Columns ! Select an item...

Include columns pkinstituicao

Tablespace txnomeinstituicao

Fill factor fklogradouro

Deferred txcomplemento

A tela de escolha do campo fica assim e ao escolher e preencher os outros campos a tela fica como mostrada a seguir:

General Definition

Columns pkinstituicao x

Include columns Select an item...

Tablespace pg_default

Fill factor

Deferrable? ☐

Deferred? ☐

NOTE: NÃO foram marcados os campos DEFERRABLE e DEFERRED, mais tarde isso será explicado. Por enquanto, NÃO marque estes itens.

Agora basta clicar no botão SAVE e sua PK está criada.

É preciso indicar uma situação importante: A CRIAÇÃO das Chaves Estrangeiras (FKs) DEPENDE da CRIAÇÃO das correspondentes Chaves Primárias. POR ISSO: CRIE TODAS AS CHAVES PRIMÁRIAS DE TODAS AS TABELAS DA BASE DE DADOS.

Agora vamos iniciar a criação das Chaves Estrangeiras.

Chaves Estrangeiras

Do mesmo modo como se inicia a edição das chaves primárias das tabelas, acessamos o a tela de definição das chaves estrangeiras marcando a tabela e acessando o menu POPUP com o botão secundário do mouse e buscando o item 'PROPERTIES'. Acessando a aba CONSTATRINTS e a sub-aba FOREIGN KEYS, vemos:

Name	Columns	Referenced Table
------	---------	------------------

Clicando no sinal “+” do lado direito, vemos:

Name	Columns	Referenced Table

Please specify columns for Foreign key.

Close Reset Save

Do mesmo modo como nas PKs, as FKs não podem ter nomes repetidos na base de dados. Por isso, é aconselhável se atrelar o nome da FK ao nome da tabela e usar algum modo sequencial de identificar a FK. Neste exemplo adotamos: o nome da tabela+'fk'+n onde o 'n' assume valores numéricos. Então preenchendo a tela e a seguir clicando no ícone de edição de detalhes da FK, temos:

Name	Columns	Referenced Table
cursosfk1		

General Definition Columns Action

Name cursosfk1

Comment

Do mesmo modo como na PK, é aconselhável preencher o campo de comentário com um texto explicando a FK. Clicando na aba: DEFINITION, vemos:

Name	Columns	Referenced Table
cursosfk1		

General
Definition
Columns
Action

Deferrable? ☒
Deferred? ☒
Match type: FULL SIMPLE (selected)
Validated? ☐
Auto FK index? ☒
Covering index: icursosfk1

Marque os itens DEFERRABLE e DEFERRED. Marque também o campo “AUTO FK INDEX” e no campo nome escreva o nome do índice que será criado ao criar esta FK. Note: o nome do índice é formado com a letra “i” seguido do nome da FK. Simples assim! O tipo de encaixe dos valores da FK podem ser escolhido entre FULL e SIMPLE. DEIXE MARCADO o item SIMPLE.

Na sub-aba COLUMNS se faz a escolha do campo da tabela em edição e para qual campo (como PK) será referenciada a FK que se está criando. Acessando então esta sub-aba:

General
Definition
Columns
Action

Columns

Local column: Select an item...
References: Select an item...
Referencing: Select an item...
Add

Local	Referenced	Referenced Table
-------	------------	------------------

Os itens desta tela são:

Local column → Nome do campo que será associado na FK que se está definindo.

References → Nome da tabela onde está a PK correspondente.

Referencing → Nome do campo que é referenciado pela FK.

Os campos devem ser preenchidos clicando na caixa de seleção correspondente. A única sequencia exigida é REFERENCES e DEPOIS o REFERENCING. Uma vez que a lista de REFERENCING é ‘montada’ depois que se escolhe a tabela onde está a PK. No exemplo a tela preenchida deve ficar assim:

Columns

Local column: fkinstituicao
References: public.instituicoesensino
Referencing: pkinstituicao
Add

Local	Referenced	Referenced Table
-------	------------	------------------

A referência está montada, agora clica-se no botão ADD que está à Direita. Depois disso a referência fica informada.

Então ao clicar no botão ADD, temos:

	Name	Columns	Referenced Table
 	cursosfk1	(fkinstituicao) -> (pkinstituicao)	public.instituicoesensino

General

Definition

Columns

Action

Columns

Local column

Select an item... | v

References

 public.instituicoesensino

Referencing

Select an item... | v

Add

	Local	Referenced	Referenced Table
	fkinstituicao	pkinstituicao	public.instituicoesensino

Clicando em Action, vemos:

	Name	Columns	Referenced Table
 	cursosfk1	(fkinstituicao) -> (pkinstituicao)	public.instituicoesensino

General

Definition

Columns

Action

On update

NO ACTION | v

On delete

NO ACTION x | v

Falta informar as ações decorrentes na tabela que tem a FK quando registros relativos à PK tiverem o valor da PK ALTERADOS ou forem EXCLUÍDOS. Estas ações são então relativas à 'ON UPDATE' e 'ON DELETE'. No caso do PostgreSQL, as ações são:

NO ACTION: os valores da FK são mantidos.

SET NULL: os valores da FK são alterados para NULL.

CASCADE: a ação feita na PK é repetida na FK.

RESTRICT: ANTES da ação na PK, se existir algum valor na FK a ação na PK NÃO É FEITA.

SET DEFAULT: Quando se cria o campo na tabela é possível definir valor padrão para o campo (caso não seja informado em um comando). Então este valor padrão será assumido na ação decorrente na FK caso a PK altere ou seja deletada.

Este último tipo de ação existe implementado em outros SGBDs, porém não em todos.

Escolhido o conjunto de ações podemos ter, neste exemplo, a tela como apresentado abaixo.

Name	Columns	Referenced Table
  cursosfk1	(fkinstituicao) -> (pkinstituicao)	public.instituicoesensino

General

Definition

Columns

Action

On update

CASCADE

| v

On delete

RESTRICT

x | v

O PGAdmin 4 é uma boa ferramenta para facilitar a construção de comandos em SQL que são executados no PostgreSQL. Para visualizar o comando construído basta, à qualquer momento, acessar a aba SQL. No exemplo deste artigo, na aba temos:

 cursos  

General

Columns

Advanced

Constraints

Parameters

Security

SQL

```
1 ALTER TABLE IF EXISTS public.cursos
2   ADD CONSTRAINT cursosfk1 FOREIGN KEY (fkinstituicao)
3   REFERENCES public.instituicoesensino (pkinstituicao) MATCH SIMPLE
4   ON UPDATE CASCADE
5   ON DELETE RESTRICT
6   DEFERRABLE INITIALLY DEFERRED
7   NOT VALID;
8 CREATE INDEX IF NOT EXISTS icursosfk1
9   ON public.cursos(fkinstituicao);
```

Close

Reset

Save

Para gravar esta definição basta o clique no botão SAVE.

Encerro este artigo aqui.