



Trabalhos

[Jump to bottom](#)

Igor Avila Pereira edited this page 18 minutes ago · 5 revisions

Trabalho 2

Construa um B.D de gerenciamento de faxinas para diaristas:

O B.D deve controlar o histórico de faxinas realizados por cada diarista em cada residência.

- Cada **Diarista** possui um identificador (**id**), **cpf** e **nome**;
- Cada **Residência** possui um **responsável**. Para o responsável é importante armazenar seu **nome** e seu **cpf**;
 - Além do responsável, cada Residência possui um **identificador** (**id**), **cidade**, **bairro**, **rua**, **complemento** e **número** e **Tamanho** (pequena, média ou grande);
- Um **Responsável** pode ser responsável por mais de uma **Residência** mas uma **Residência** tem somente um **Responsável**;
- Dependendo do **Tamanho** da **Residência**, a **Faxina** tem um **preço**;
- Uma **Diarista** realiza uma **Faxina** por dia, ou seja, atende uma Residência por dia;
- Uma **Diarista** pode atender várias residências, e uma residência pode ser atendida por várias **Diaristas**;
- **Faxinas** podem ser agendadas (por data);
- É importante saber se a **Diarista** não foi (faltou) a uma **Faxina**, previamente, agendada;
- **Faxinas** agendadas e não-realizadas (por qualquer motivo) não devem ser pagas;
- É importante armazenar feedbacks de avaliação por cada **Faxina** realizada. Estes comentários devem ser realizados pelo **Responsável** da **Residência** no momento da conclusão da **Faxina**;
- O valor final pago pela **Faxina** deve ser também armazenado pois o valor por ser: **maior** (devido à gorjetas), **menor** (devido à algum dano/prejuízo causado pela diarista) ou **igual** ao valor definido para residências de mesmo **Tamanho**. Lembrando que, ao longo do tempo, o valor da **Faxina** atribuído pelo **Tamanho** da **Residência** pode também mudar. Logo, é importante armazenar também este valor que foi realmente pago por cada **Faxina** a fim de saber, especificamente, o que cada **Diarista** recebeu pelas **Faxinas** que fez ao longo do tempo.



Copied!

Exigências:

- Crie o Modelo Relacional
- Implemente no PostgreSQL o B.D projetado no Modelo Relacional (construa um *script.sql*)
- Crie um STORE PROCEDURE que permita agendar quinzenalmente ou mensalmente faxinas em uma determinada residência:
- A diarista e a residência devem ser considerados parâmetros de entrada. Por outro lado, pode-se realizar o stored procedure de 2 formas: Opções: 1) Utilizar uma data limite (ex: até 31/12 do ano atual) 2) Utilizar uma quantidade máxima de agendamentos (ex: marcar 30 faxinas mensalmente)
- Crie um STORE PROCEDURE que calcule a porcentagem de presenças que uma diarista obteve em suas faxinas ao longo do ano:
 - Ex: 75% de presença
- Crie uma TRIGGER que exclua a diarista caso suas presenças fiquem menores que 75% (quando a diarista já tem no mínimo 5 faxinas cadastradas)

Valor: 5

Trabalho 1

Em um sistema de blogs, cada **Post** possui um **identificador único**, um **título**, um **texto** e uma **data/hora** de publicação.

Cada **Post** pode ser escrito por diversos **autores** que, por sua vez, pode escrever diversos **posts**. No entanto, quando um post é criado, define-se se o mesmo pode ser compartilhado com os demais autores da plataforma, ou seja, estabelece se mais de um autor possa participar de sua edição - ou fica exclusivo somente para autor/criador. Cada autor tem um **identificador único**, um **nome**, um **email** e uma **senha** de acesso.

Além dos autores dos posts, há **leitores** que somente podem ter acesso aos **posts** (publicações) por meio de um cadastro prévio. Cada **leitor** possui as mesmas informações que os **autores** mas possuem **endereço** - usados para o envio de correspondências promocionais. Um **leitor** pode cadastrar diversos endereços. Por sua vez, um endereço pertence, exclusivamente, a um **leitor**. Cada **endereço** possui **identificador único**, um **bairro**, uma **rua**, um **número**, um **complemento** e um **cep**.

Um modelo relacional para o sistema de blogs foi proposto conforme abaixo:

Copied!

