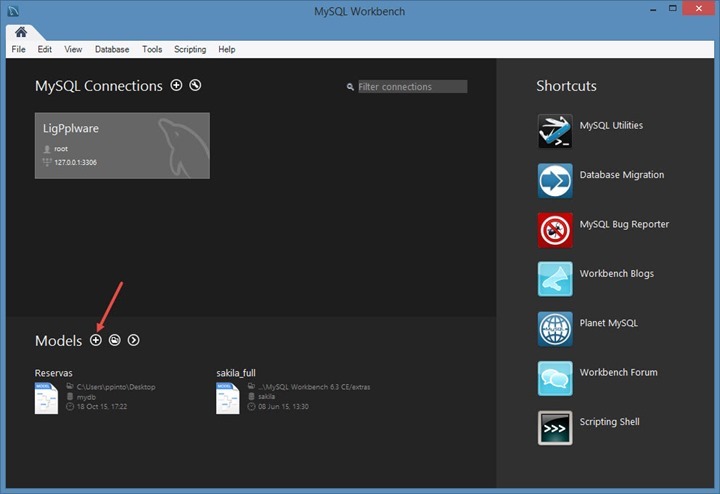
A criação pode ser feita de um dos três modos:

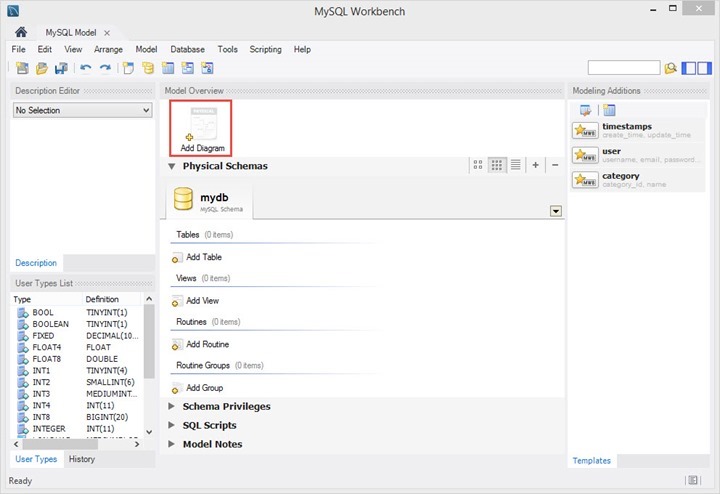
* Criar um novo diagrama EER
* Criar a partir de uma base de dados existente
* Criar a partir de um script

##### **Criar um novo diagrama EER**

Clicar no ‘+’ junto a Models

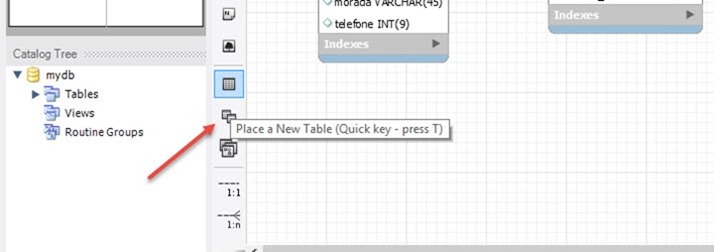
**[](https://pplware.sapo.pt/wp-content/uploads/2015/10/eer_00.jpg)**

Clicar em em “Add Diagram”

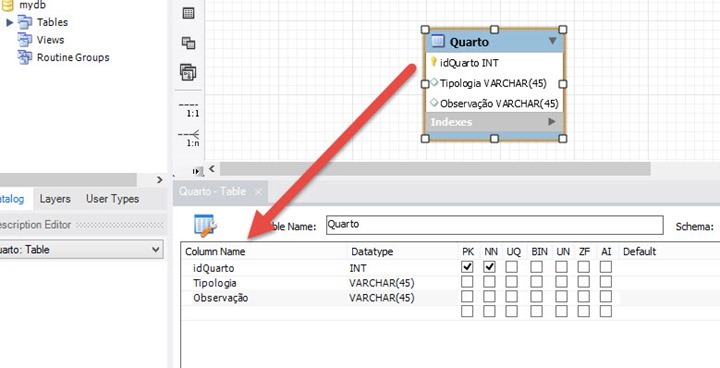
**[](https://pplware.sapo.pt/wp-content/uploads/2015/10/eer_01.jpg)**

Estabelecer as respectivas relações (1:1, 1:N, N:M).

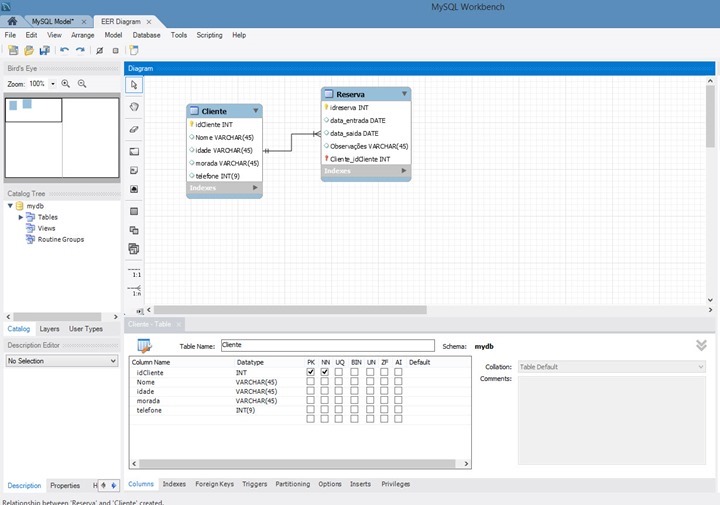
Para criar tabelas basta carregar no seguinte ícone na barra lateral.

**[](https://pplware.sapo.pt/wp-content/uploads/2015/10/eer_03.jpg)**

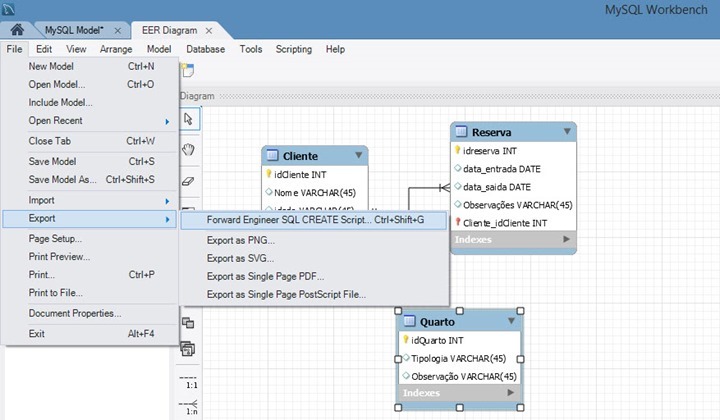
Indicar um nome para a tabela (ex. Quarto) e definir quais os campos que fazem parte dessa tabela.

**[](https://pplware.sapo.pt/wp-content/uploads/2015/10/eer_04.jpg)**

Por fim devem **definir os relacionamentos entre tabelas**. As ligações estão também na barra lateral esquerda. O resultado final será algo semelhante ao apresentado em baixo.

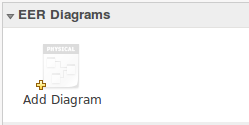
**[](https://pplware.sapo.pt/wp-content/uploads/2015/10/eer_02.jpg)**

Depois de criado o diagrama EER é possível exportá-lo para um script SQL ou até inseri-lo direto no SGBD.

**[](https://pplware.sapo.pt/wp-content/uploads/2015/10/eer_05.jpg)**

E está feito. Esta ferramenta é extremamente poderosa, com funcionalidades para quase tudo. Para quem precisa de fazer diagramas ER ou até mesmo EER aqui fica a nossa sugestão.

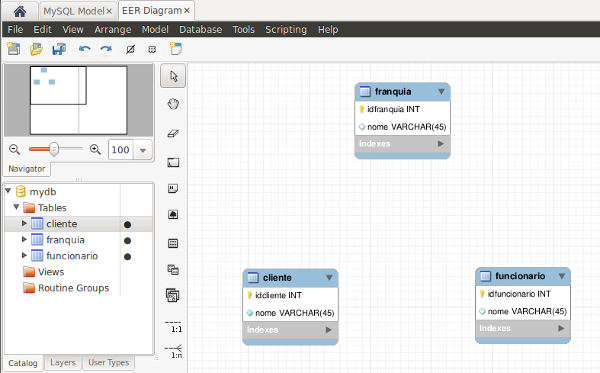
Crie um diagrama novo. Com ele, nós vamos visualizar graficamente os relacionamentos da tabela.



Trabalhando com o diagrama

Nesse momento, temos um **diagrama vazio**.

Usando o catálogo, e arrastando as tabelas que acabamos de criar, teremos o seguinte esquema (sem nenhum relacionamento).



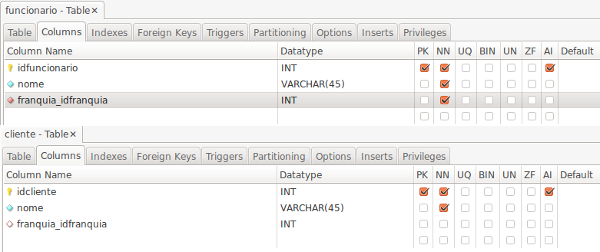
À partir de agora, veremos a **grande vantagem** do **MySQL workbench**. Use as ferramentas para criar os relacionamentos.

mysql-workbench-tools.png

Use a primeira 1:n, e então **clique em cliente e depois em franquia**. Com isso, você criou o primeiro relacionamento, em que cliente se relaciona com franquia através de uma coluna (no caso, chamada de "franquia\_idfranquia").

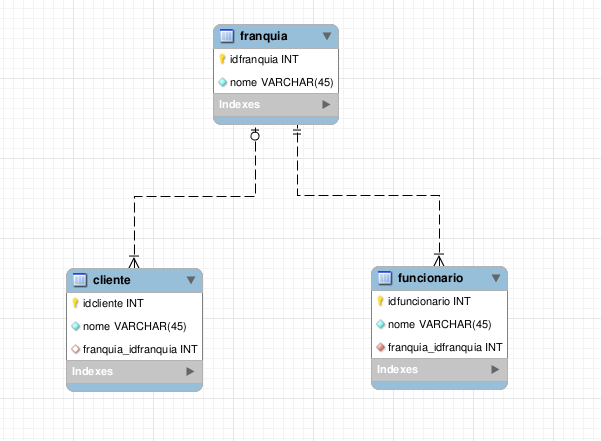
Agora use a segunda 1:n, e então **clique em funcionário e depois em franquia**. Agora, criamos outro relacionamento, porém esse identifica o funcionário como se fosse dessa franquia em específico.

Além disso, eu prefiro fazer outras coisas. Edita as colunas da seguinte forma:



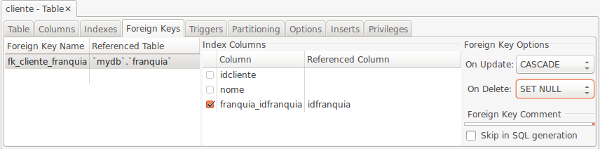
Eu fiz isso dessa forma, pois quero habilitar o NULL apenas para o cliente (pois tenho a intenção de haver cliente sem empresa), e não tenho a necessidade de que franquia\_idfranquia seja identificador do funcionário.

Se estamos no mesmo passo, seu **novo diagrama** deve ser semelhante ao seguinte:



Configurando os relacionamentos

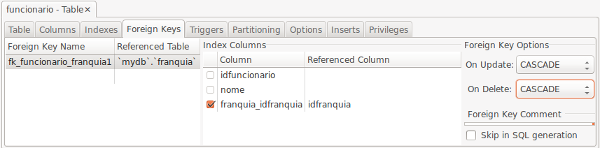
Estamos no final. Selecione a **tabela de clientes**, e a aba de Foreign Keys. Em **Foreign Key Options**, deixe **CASCADE** para Update e **SET NULL** para Delete.



Com isso, você diz para o BD que ao atualizarmos franquia, ele vai atualizar os clientes também. Porém, ao apagá-lo, você vai deixar os clientes com franquia desconhecida (franquia = NULL). É possível encontrá-los depois com uma query simples:

SELECT \* FROM `cliente` WHERE franquia\_idfranquia IS NULL

E por fim, configurar **funcionario** com **CASCADE** para ambos.



Com isso, ao atualizar ou apagar a franquia, você fará o mesmo com seus funcionários.