
LABORATÓRIO DE
BANCO DE DADOS
IBD100

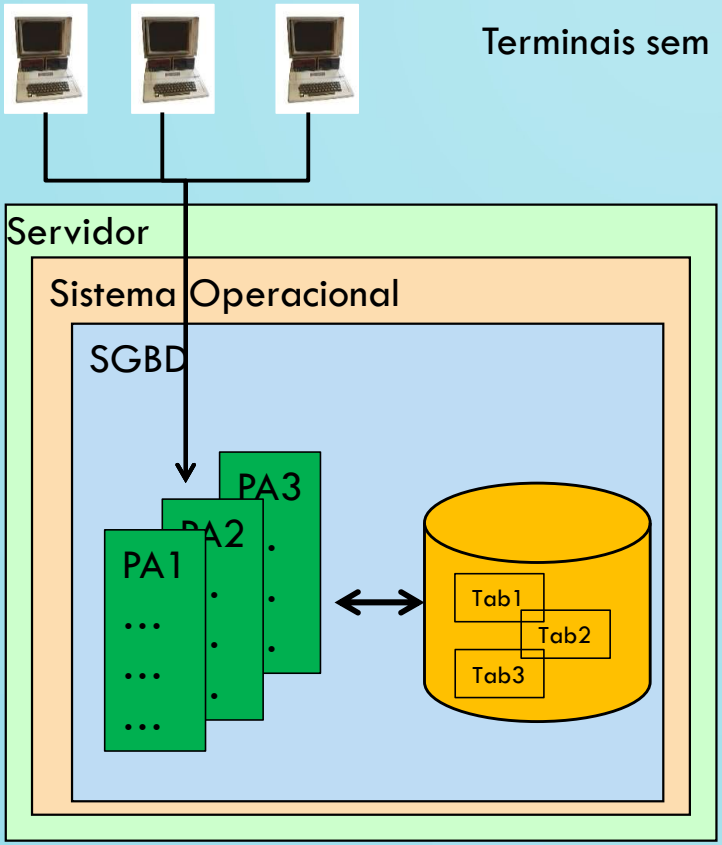
IBD100
TRIGGERS E STORED-PROCEDURES

João Maurício Hypólito
prof.oao.hypolito@gmail.com

Programa

- Histórico dos SGBD Distribuídos
  - Origem (motivo)
  - Soluções interessantes
  - Mas tinham alguns problemas
  - Uma Solução (para alguns problemas)
- Definindo Gatilhos
  - Formas de implementação
  - O PostgreSQL implementa como?
  - Um pouco da PLpgSQL
  - Um exemplo de escrita de uma função de gatilho
- Os gatilhos que devem ser feitos
  - Como resolver? Método!

- A evolução dos SGBDs em 40 anos (de TI para TIC)
  - As máquinas tinham a arquitetura centralizada (40-80).

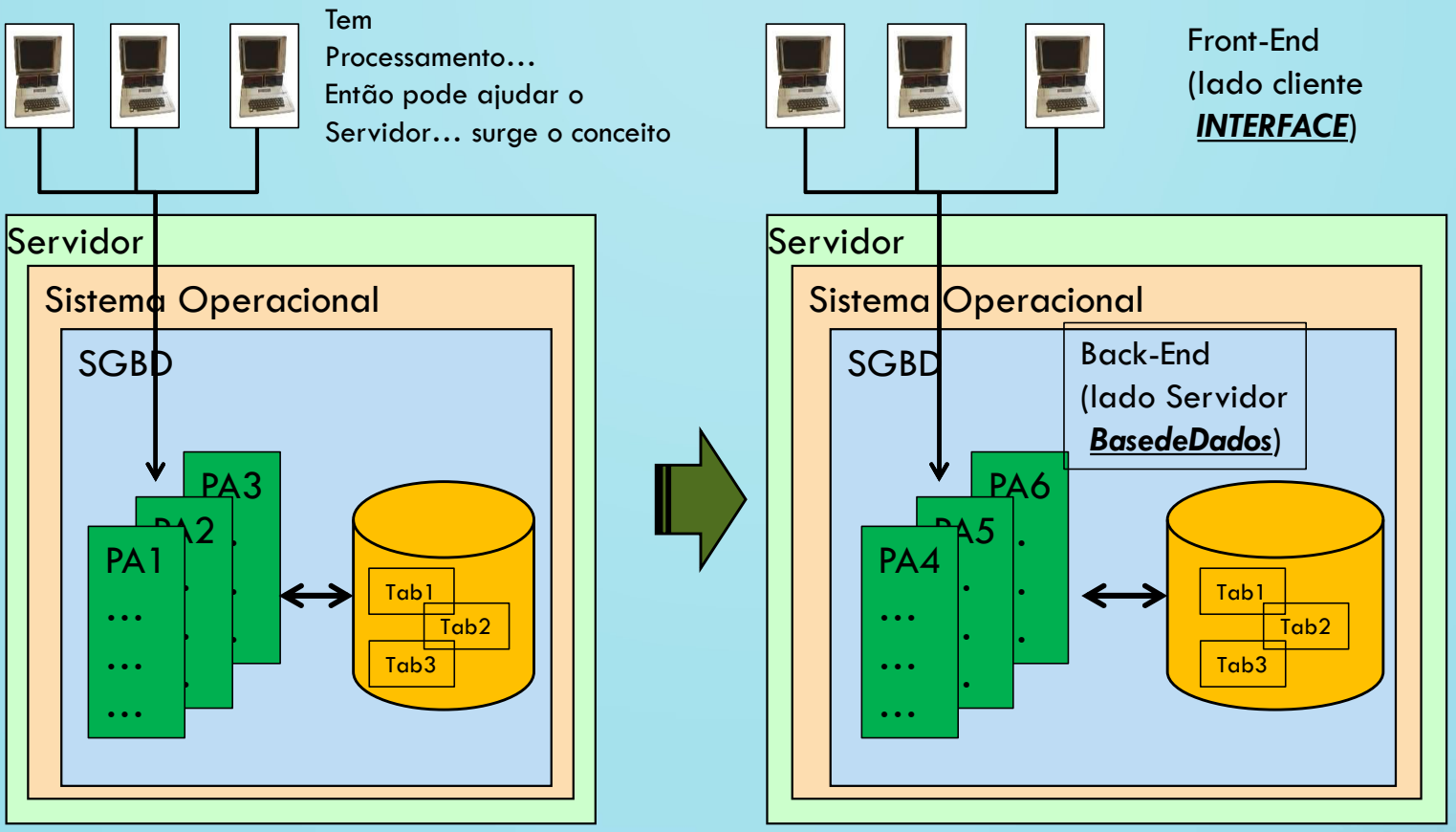


Terminais sem processamento

Todo processamento é oferecido em uma única máquina: Mainframes
A administração é facilitada mas o recurso se tornou limitado (houve um aumento rápido da procura por poder computacional).

João Maurício Hypólito - prof.joao.hypolito@gmail.com

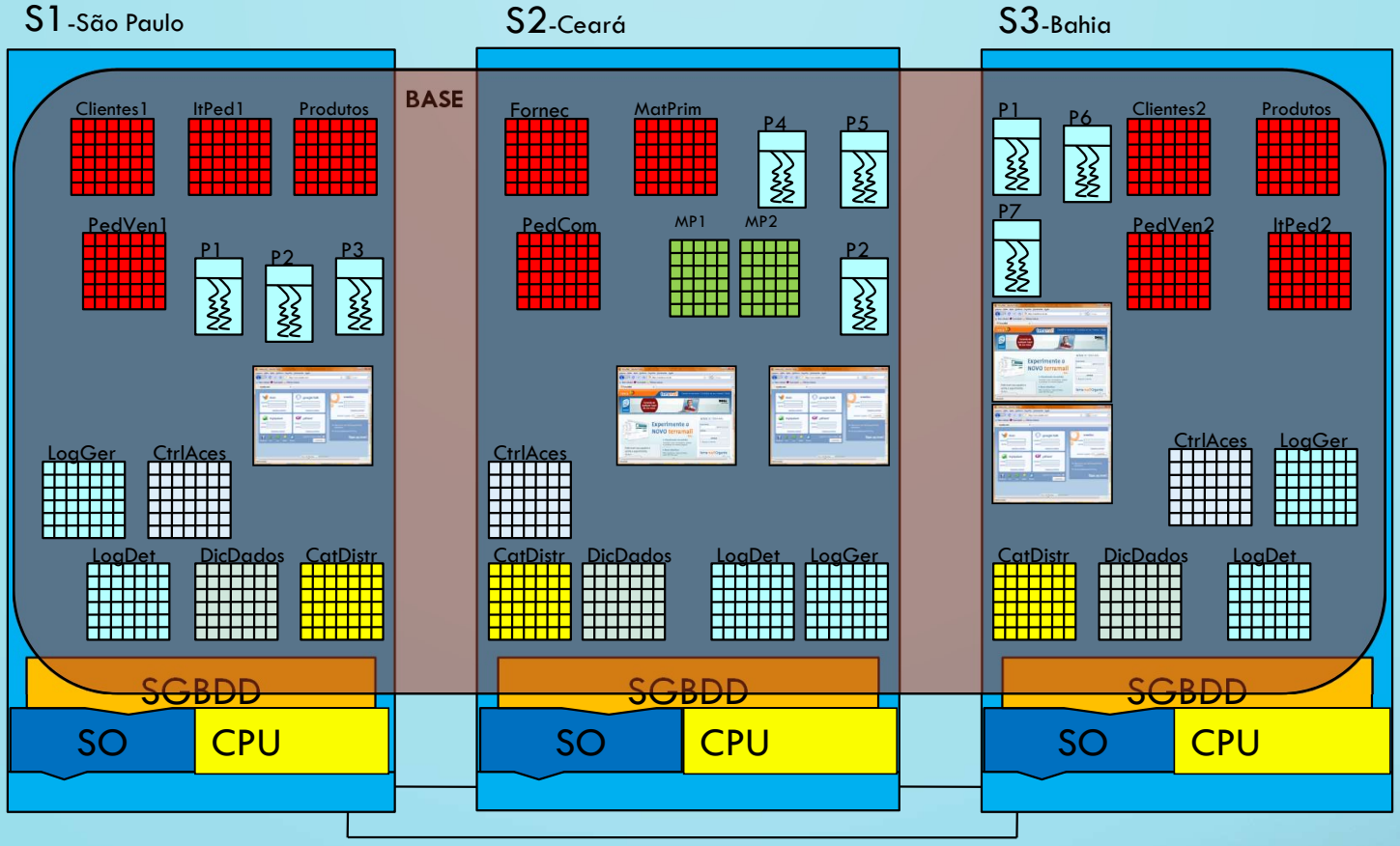
- A evolução dos SGBDs em 40 anos (de TI para TIC)
  - Mas vieram as Redes de Computadores (80-90)



João Maurício Hypólito - prof.joao.hypolito@gmail.com

IBD100 – GATILHOS (TRIGGERS E STORED-PROCEDURES)

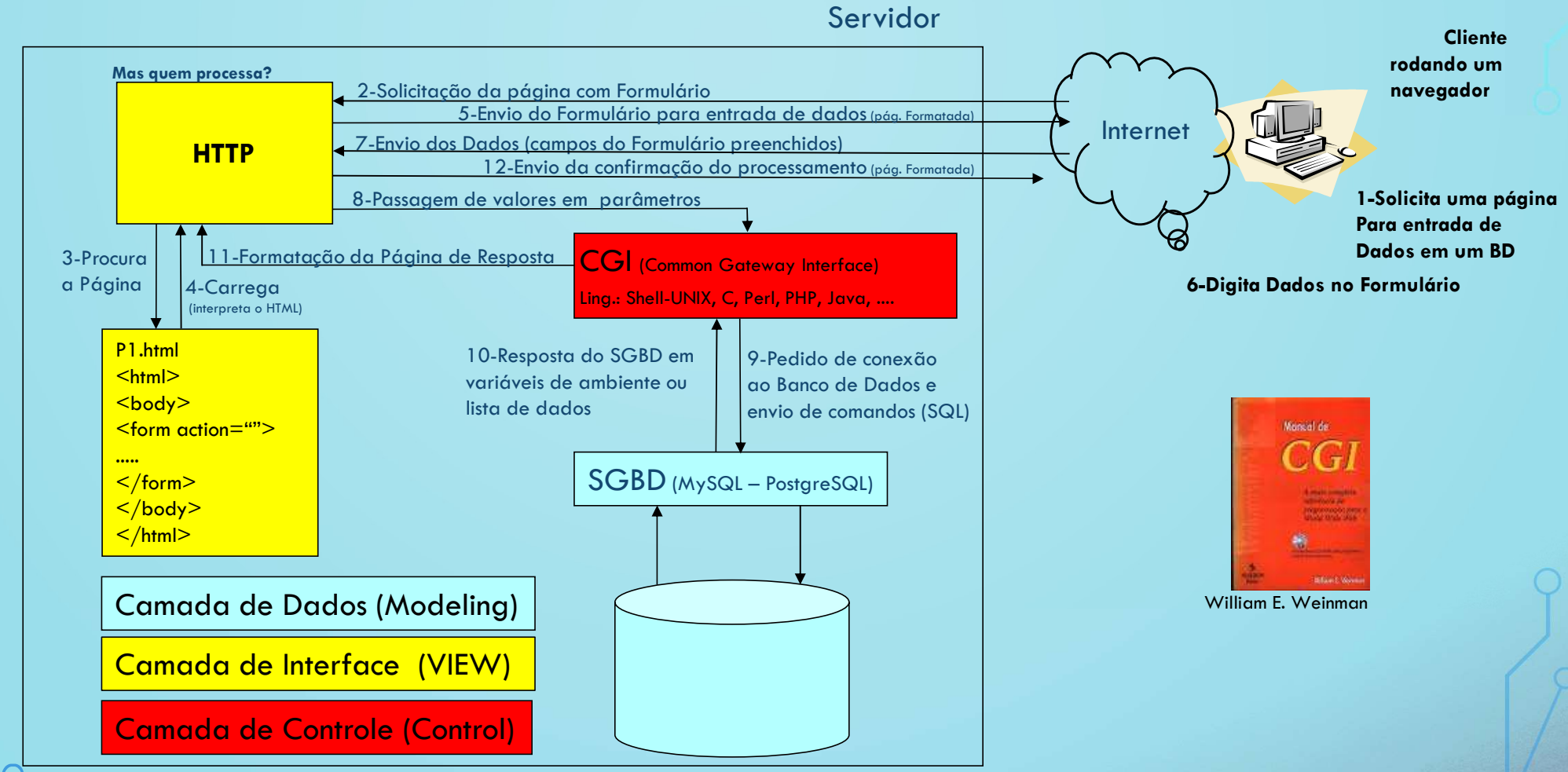
- Os Bancos de Dados Distribuídos
  - O Conceito Básico: O usuário final, analistas e programadores NÃO sabem onde se hospedam os componentes da base de dados.



João Maurício Hypólito - prof.joao.hypolito@gmail.com

IBD100 – GATILHOS (TRIGGERS E STORED-PROCEDURES)

- Finalmente, tudo converge para o separado e especializado (cada um é muito bom no que sabe fazer). É o princípio da convergência e transparência.



William E. Weinman

João Maurício Hypólito

prof.joao.hypolito@gmail.com

IBD100 – GATILHOS – PROBLEMAS DAS TECNOLOGIAS EMERGENTES

- Durante o desenvolvimento da tecnologia de Bancos de Dados Distribuídos uma série de problemas não apresentavam solução satisfatória:
  - Como controlar o Desenvolvimento de Aplicações sobre bancos de dados em um contexto que deveria atender a diferentes áreas funcionais de uma organização?
  - Como poder garantir que PA feitos em uma AF respeitassem as regras do negócio da organização?
  - Como evitar que as estações de trabalho fossem sobrecarregadas com o processamento de dados?
- Estes problemas aumentaram com o estudo e desenvolvimento de serviços e produtos na estrutura MVC.
- Os desenvolvedores de cada camada passaram a trabalhar em suas próprias soluções acelerando o desenvolvimento.
- Na área de ferramentas para bancos de dados...

João Maurício Hypólito - prof.joao.hypolito@gmail.com

IBD100 – GATILHOS - SOLUÇÃO NA ÁREA DE BD

- Outra solução foi o desenvolvimento de SGBDs que permitiam que segmentos de códigos fossem anexados à estrutura das tabelas de modo que, quando realizados eventos (IEA ou IDU) eram disparados a execução dos segmentos de código.

- Trechos de **programas** associados aos arquivos de dados e **disparados** quando ocorrem eventos (Inc,Exc,Alt).
  - Então pensamos em 'programar' o banco de dados, ou ainda em 'grudar' as regras nos dados: "É método de Classe?"... NÃO!
- Podem ser denominados de *triggers, stored-procedures, constraints* ou *code rule* (aí depende do SGBD).
  - Nessa tecnologia os SGBDs andaram na frente dos Padrões (de SQL). Por isso, existem muitas formas de implementar o mesmo conceito.
- As tabelas passam a **conter** as regras de negócios
  - Conceito muito próximo de Objetos (Dados+Processos)...
    Mas cuidado. Tabela não É Classe. Classe existe ↔ existem Atributos & Métodos. A Tabela existe sem métodos.

- Quem implementa: DBA.
  - DBA?? Quebra as atividades dos perfis profissionais envolvidos no projeto de Sis.
    - O DBA passa a 'programar' as regras de negócio 'grudados' aos dados das tabelas
  - Vantagem:
    - Concentra as atividades e diminui a margem de erros (sobrecarrega os DBAs, mas quem se importa?)
  - Desvantagem:
    - Concentra as atividades e desloca o foco das atividades do DBA para o negócio da empresa.

João Maurício Hypólito - prof.joao.hypolito@gmail.com

IBD100 – GATILHOS - TIPOS BÁSICOS

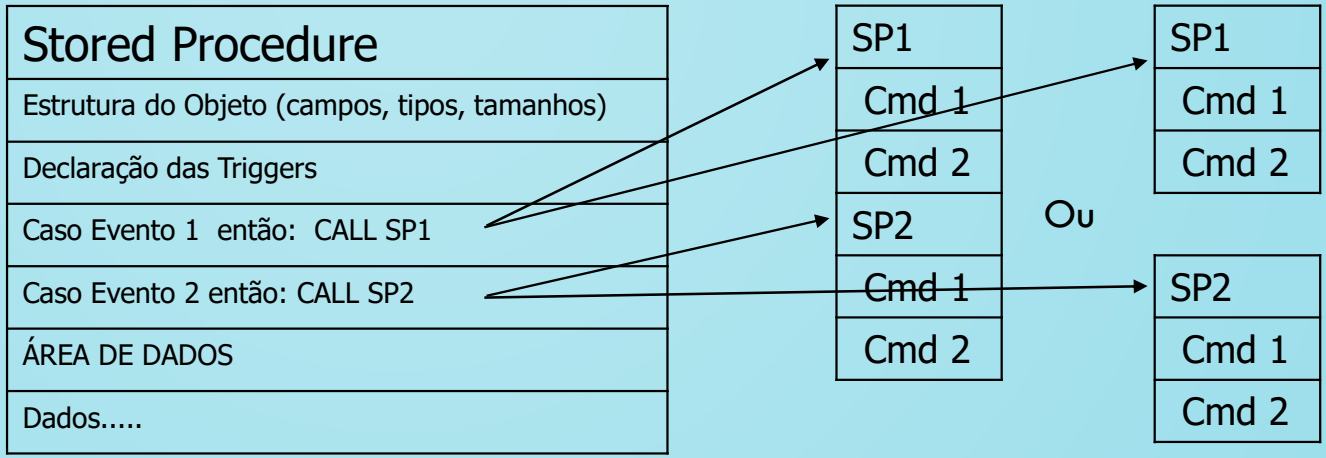
- Trigger - A regra de negócio está no mesmo arquivo que acomoda o dado.
- Vantagem:
  - Rapidez na execução (em geral tudo vai para a RAM)
  - as regras disponíveis estão em um só arquivo
- Desvantagem:
  - Dificuldade de Manutenção:
    Para *redefinir* regras é preciso bloquear o acesso ao BD
    (pode ser 'resolvido' com um
    computador para desenvolvimento)

Triggers
Estrutura do Objeto (campos, tipos, tamanhos)
Declaração das Triggers
Caso EVENTO 1
Comando 1
Comando 2
Caso Evento 2
Comando 3
Comando 4
ÁREA DE DADOS
Dados.....
Dados ...

João Maurício Hypólito - prof.joao.hypolito@gmail.com

IBD100 – GATILHOS - TIPOS BÁSICOS

- Store Procedure: No arquivo de dados existe uma chamada para um arquivo de programa onde a regra pode ser implementada.



- Vantagem: Facilidade de Manutenção

- Desvantagem: Mais chamadas ao Sistema de arquivos do SO
  (muitos arquivos acessados limita o desempenho)

João Maurício Hypólito - prof.joao.hypolito@gmail.com

IBD100 – GATILHOS - O PADRÃO SQL E AS SOLUÇÕES DOS SGBDS

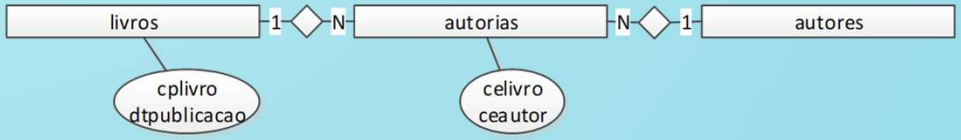
- Um pouco de história:
  - O Modelo relacional foi apresentado em 1970 (com tudo no 'combo' – incluindo AR)
  - O 'Rascunho' do SQL começou a ser estudado em 73 e em 74 já havia o SEQUEL, depois o SQUARE e a seguir o SEQUEL2 → SystemR da IBM.
  - O SQL 'aparece' em 1980 e torna-se um 'padrão de mercado' com muitos SGBDs seguindo este 'modelo de linguagem'.
  - 1986 o ANSI apresenta um padrão de SQL e é homologado pela ISO.
  - MAS o SQL já estava começando a 'descolar' da AR pois implementava novos comandos (junção aberta e processamento de funções mais complexas), daí o ANSI teve que acelerar as revisões do Padrão 'seguindo' as tendências de mercado. Vieram os padrões ANSI:92.
  - O ANSI:92 foi adotado por muitos SGBDs como o SQL referenciado, PORÉM o mundo de Banco de Dados já começava a lidar com BD DISTRIBUÍDOS. Daí o padrão novamente começou a 'ficar para trás'. Novamente o ANSI teve que estudar a forma do SQL tratar todos os avançados tecnológicos e (só em 1999) apresenta o ANSI:99 com a implementação *proposta* para a implementação de gatilhos.
  - Mas, os SGBDs já tinham implementado os seus meios para os conceitos de gatilhos. Daí que vem o 'descolamento' dos SGBDs e do Padrão ANSI.
  - Hoje, temos uma diversidade de formas de implementar gatilhos.
  - No SGBD PostgreSQL os termos de definição parecem indicar um gatilho (trigger) mas as estruturas envolvidas sugerem que seja uma Stored-Procedure.
  - De qualquer forma o conceito é o mesmo... No PostgreSQL se implementa o conceito de segmento de código executado quando se ATUALIZA uma tabela da base de dados.
  - No PostgreSQL escrevemos a codificação do gatilho no componente "TRIGGER FUNCTIONS"
  - Depois de codificados, as Trigger Functions devem ser VINCULADAS as tabelas. Isso se faz criando NA TABELA um TRIGGER que 'dispara' uma 'Trigger Function'.

João Maurício Hypólito - prof.joao.hypolito@gmail.com

IBD100 – GATILHOS - O MODO DE IMPLEMENTAÇÃO DO POSTGRESQL

- Mas... Como construir os gatilhos?
- Um gatilho dever ser sempre associado a UMA tabela (só uma, embora possa consultar dados de outras tabelas).
- Um gatilho deve ser sempre associado a um evento sobre os dados da tabela. Evento={Inclusão, Exclusão ou Alteração} (IUD).
- Um gatilho deve ser sempre associado a um momento quando deve ser executado. Em geral ANTES | DEPOIS de uma ação (Before | After).
- Exemplo:
  - Um autor não pode ter uma autoria incluída se o último livro escrito tiver sido escrito a menos de 2 anos. Lembrando que...
  - Então...
    Tabela referenciada = autorias.
    Evento = Inclusão.
    Momento = Antes da ação (Before)

    } Isso PODE indicar o nome do gatilho: **autoriasBI**



João Maurício Hypólito - prof.joao.hypolito@gmail.com

IBD100 – GATILHOS - O MODO DE IMPLEMENTAÇÃO DO POSTGRESQL

- O que deve ser feito no desenvolvimento deste gatilho?
- Pegar em uma variável (vdias) a diferença (em dias) da data de hoje menos a data da última publicação de livros do mesmo autor do registro que se está tentando incluir em autorias.
- SE o valor de vdias for menor do que 731 dias (dois anos),
  ENTÃO a autoria não pode ser incluída,
  SENÃO a autoria pode ser incluída.
- Hum...
  Teremos que declarar uma variável, ler o dado da tabela livros (dtpublicacao), fazer a comparação e retornar Ok! Ou NOk!
- Mas em qual linguagem?
  - O PG aceita que programas de gatilhos possam ser escritos em diversas linguagens.
  - Usaremos a Procedural Language do PostgreSQL (PL/pgSQL).
- Codificando...

João Maurício Hypólito - prof.joao.hypolito@gmail.com

IBD100 – GATILHOS - O MODO DE IMPLEMENTAÇÃO DO POSTGRESQL

- Codificando...

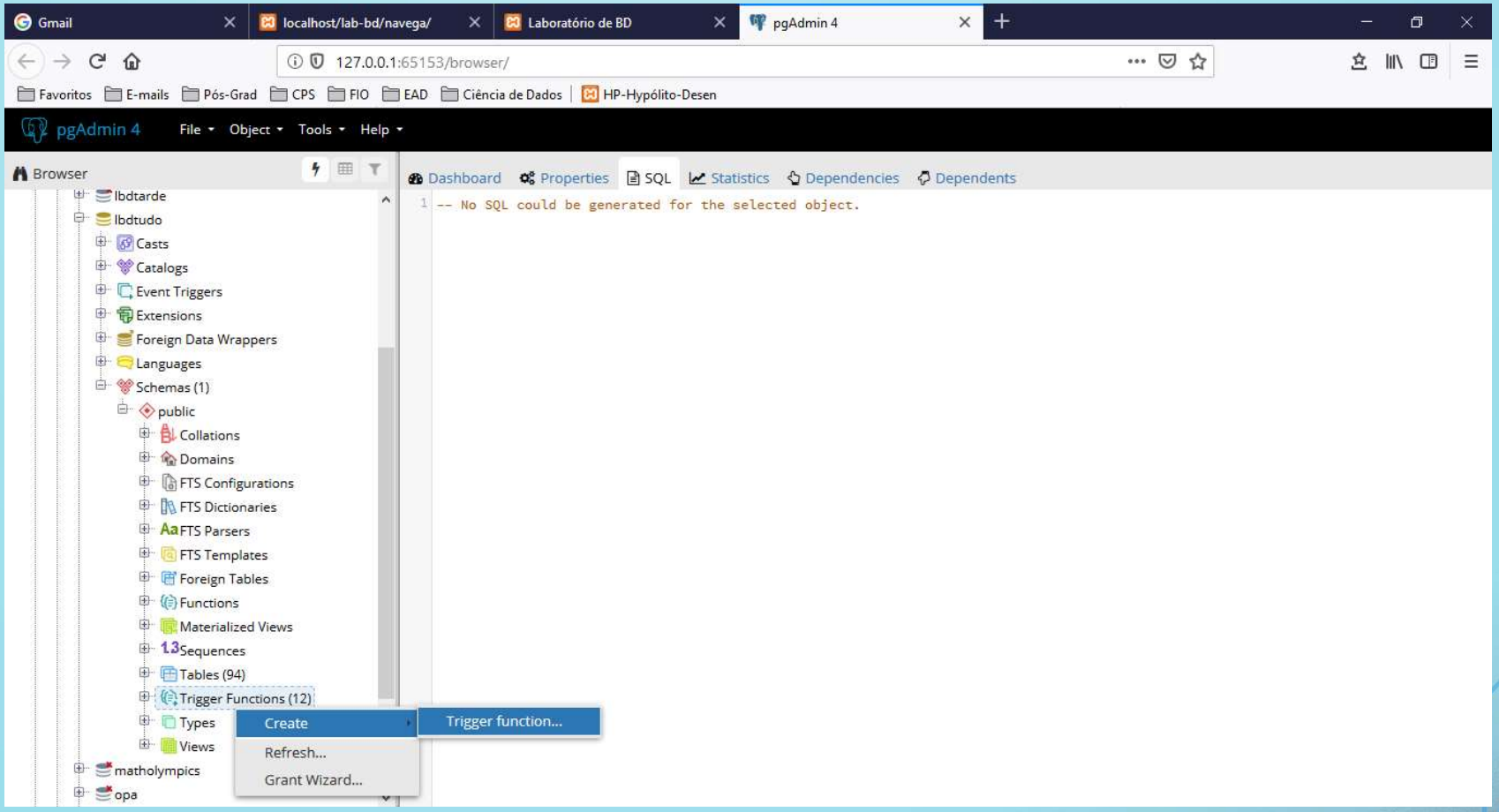
```
DECLARE
  vdias smallint;
BEGIN
  SELECT current_date-MAX(dtpublicacao) INTO vdias
              FROM livros INNER JOIN autorias ON livros.pklivro=autorias.fklivro
              WHERE autorias.fkautor=NEW.fkautor
              GROUP BY fkautor
              ORDER BY fkautor;
  IF ( vdias<731 )
  THEN
    RAISE EXCEPTION 'O Autor (%) publicou a menos de 2 anos',NEW.fkautor;
    RETURN NULL;
  ELSE
    RETURN NEW;
  END IF;
END
```

- Este código é identificado com nome: autoriasBI e é escrito como uma *FUNÇÃO* de GATILHO (*trigger function*) no PG (esse SGBD tem esta estratégia). Uma Trigger Function pode (até) ser 'colada' em mais de tabela (meio difícil disso acontecer, mas no PG *é possível*).

João Maurício Hypólito - prof.joao.hypolito@gmail.com

IBD100 – GATILHOS - O MODO DE IMPLEMENTAÇÃO DO POSTGRESQL

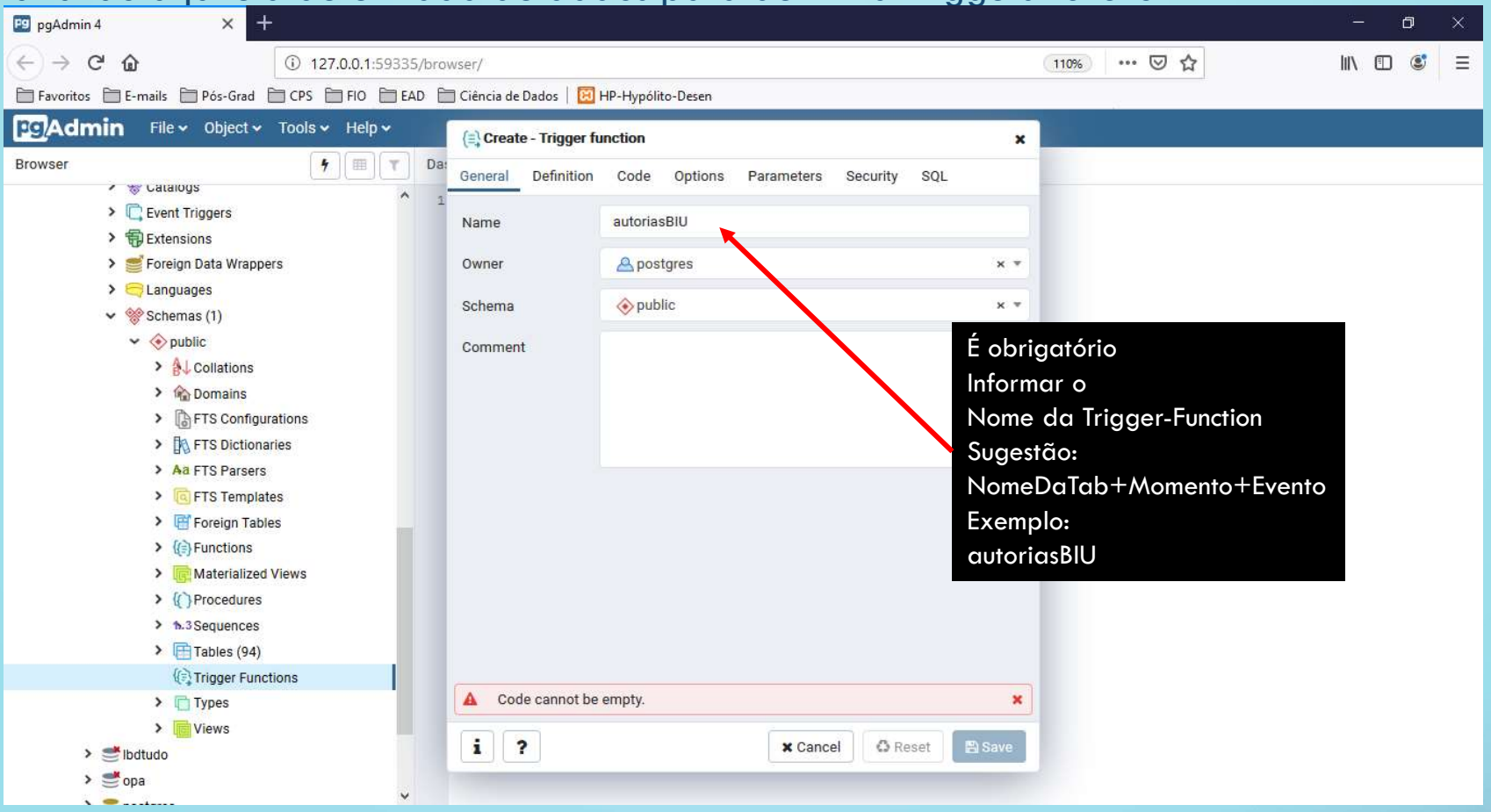
- Construindo no pgAdmin4.
- Acessando a lista de Triggers Function



João Maurício Hypólito - prof.joao.hypolito@gmail.com

IBD100 – GATILHOS - O MODO DE IMPLEMENTAÇÃO DO POSTGRESQL

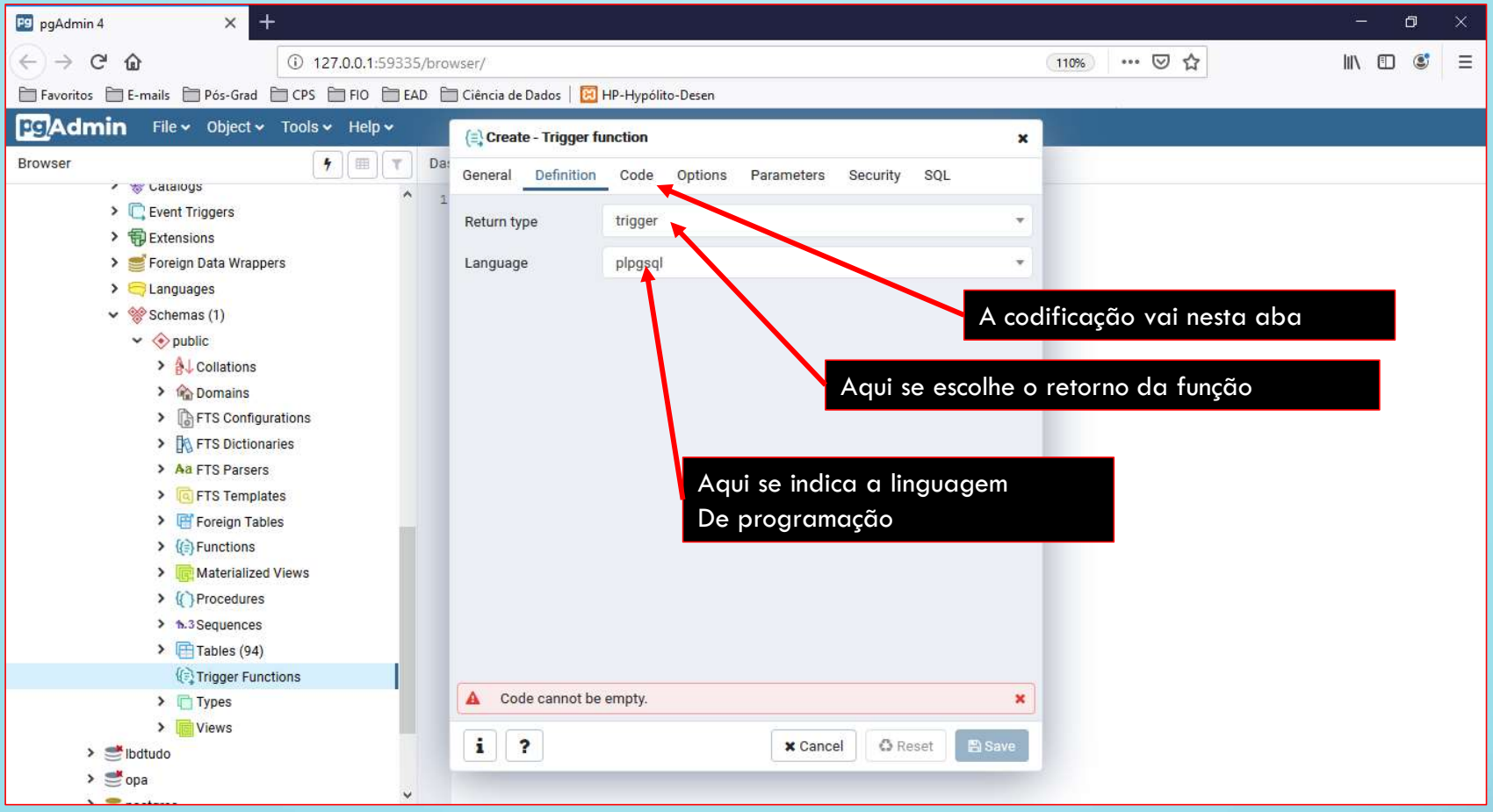
- Construindo no pgAdmin4.
- Editando a janela de entrada de dados para definir a Triggers Function



João Maurício Hypólito - prof.joao.hypolito@gmail.com

IBD100 – GATILHOS - O MODO DE IMPLEMENTAÇÃO DO POSTGRESQL

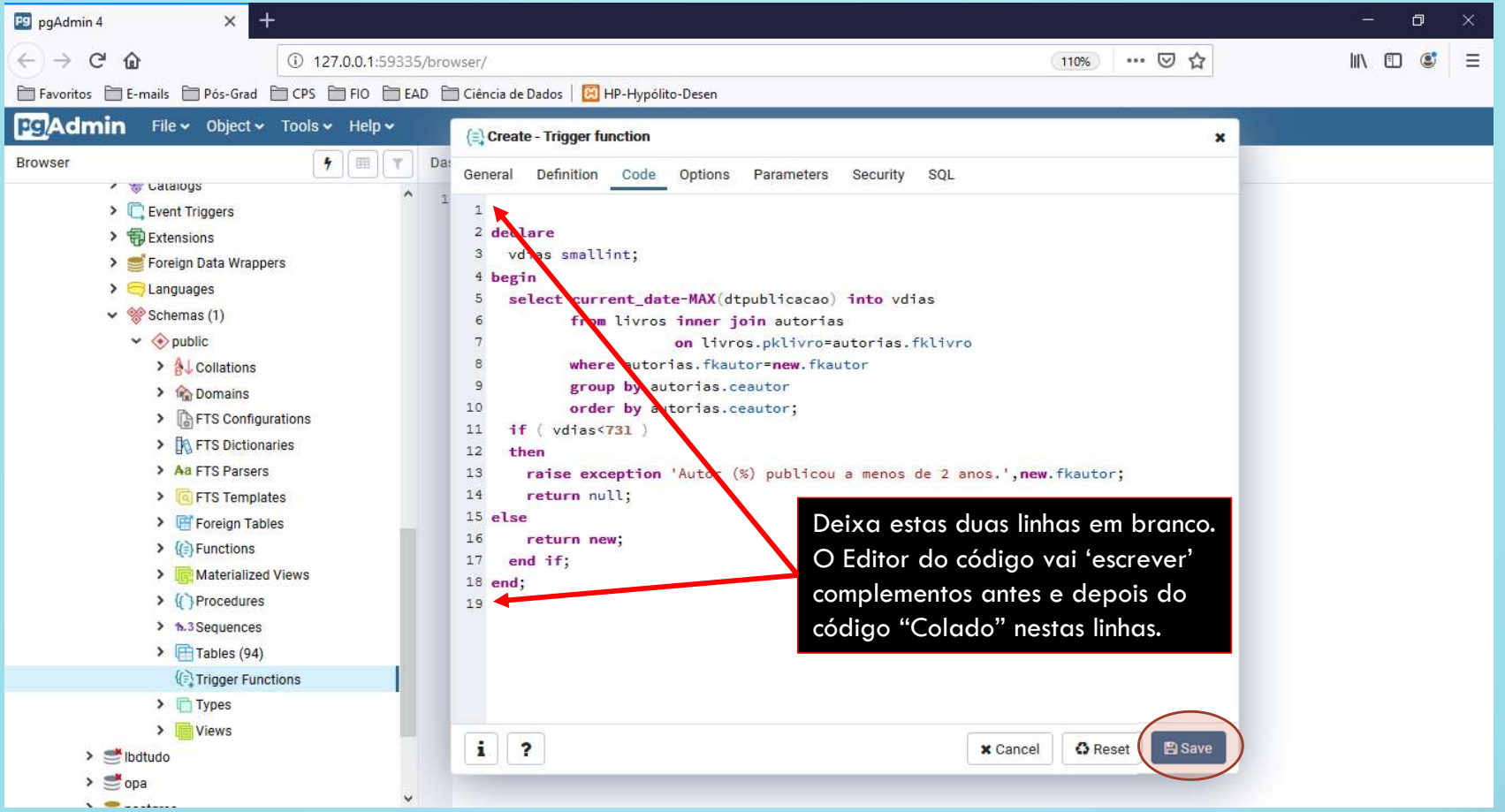
- Construindo no pgAdmin4.



João Maurício Hypólito - prof.joao.hypolito@gmail.com

IBD100 – GATILHOS - O MODO DE IMPLEMENTAÇÃO DO POSTGRESQL

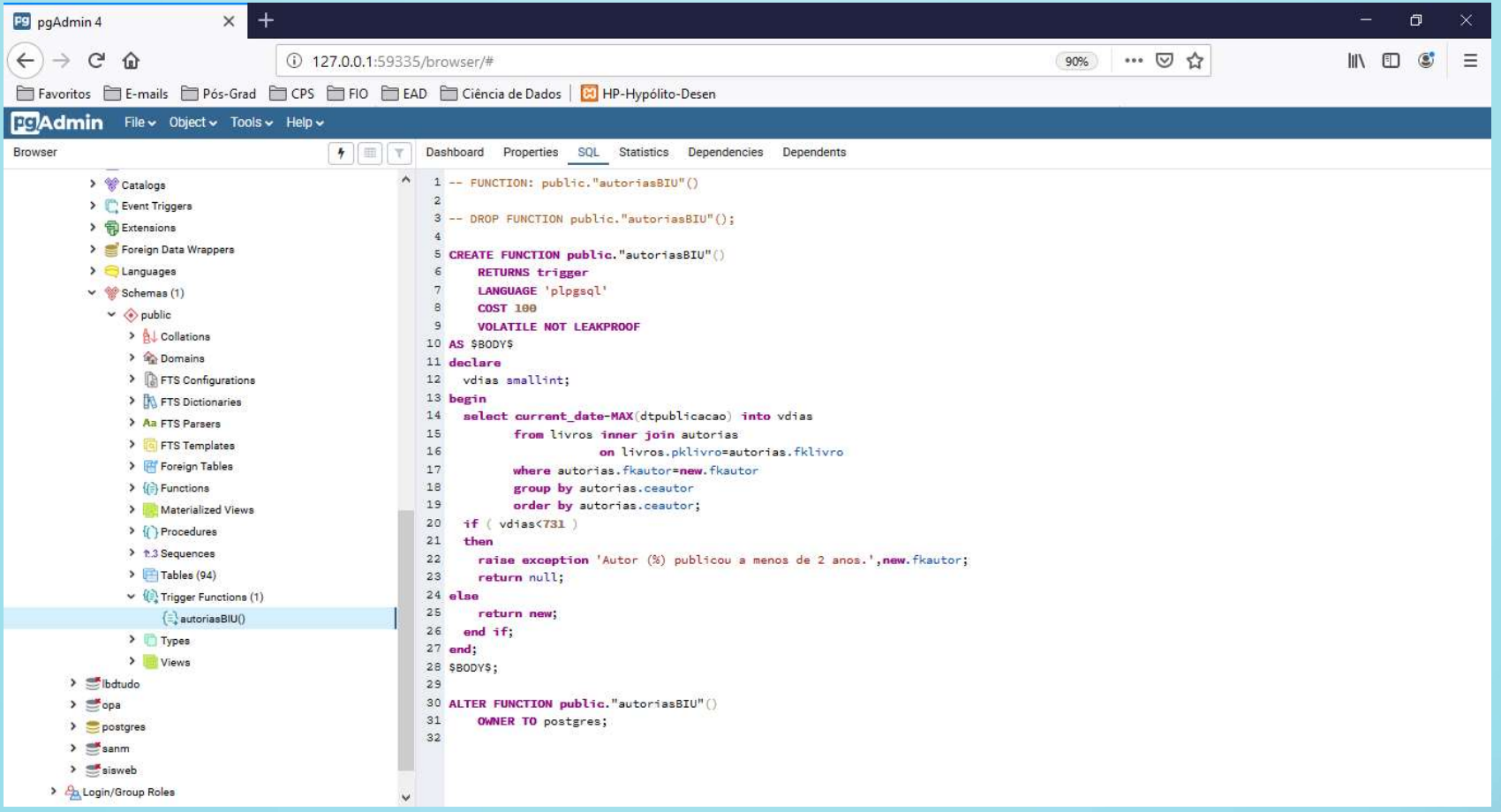
- Construindo no pgAdmin4.



João Maurício Hypólito - prof.joao.hypolito@gmail.com

IBD100 – GATILHOS - O MODO DE IMPLEMENTAÇÃO DO POSTGRESQL

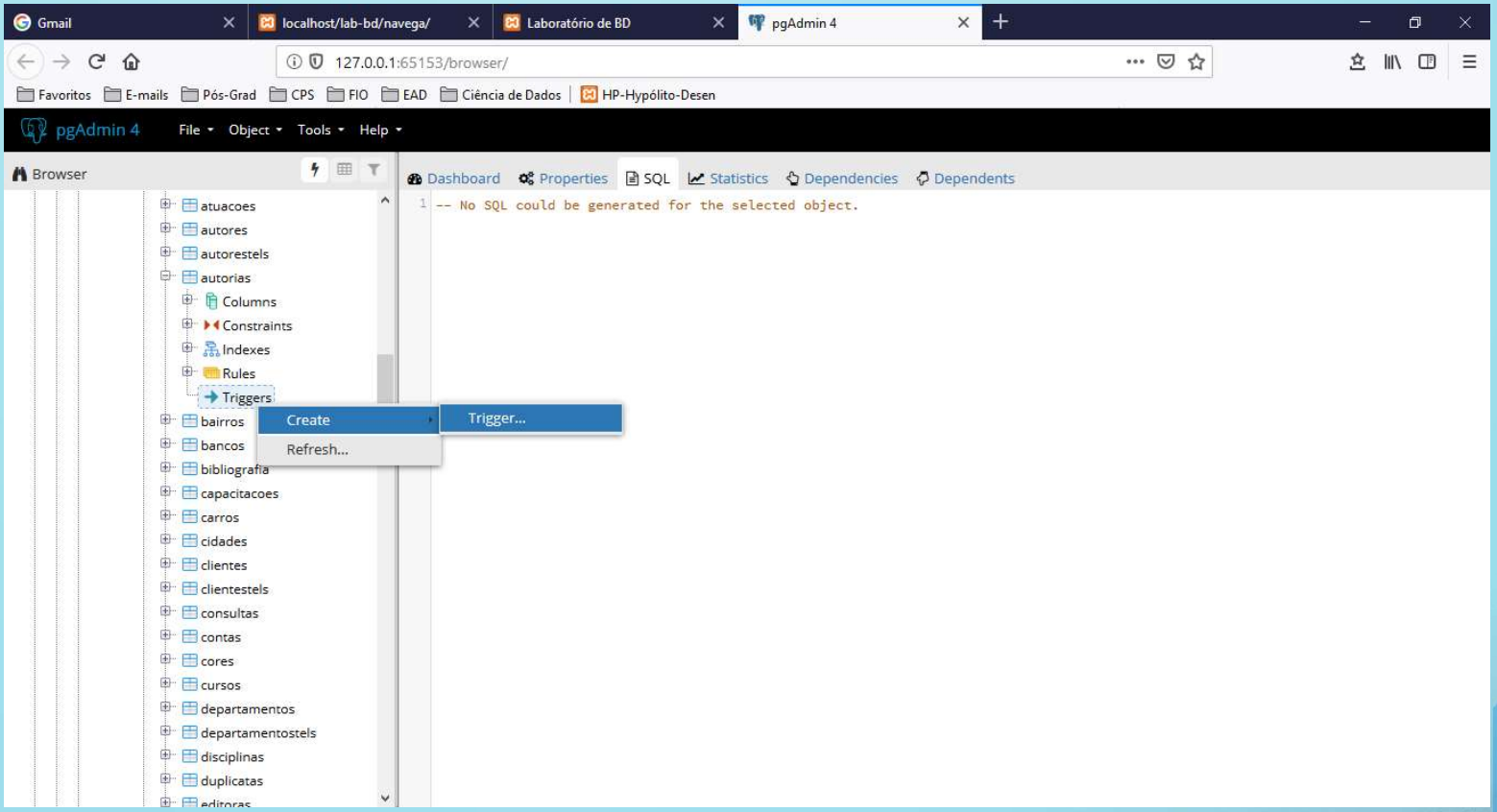
- Construindo no pgAdmin4.



João Maurício Hypólito - prof.joao.hypolito@gmail.com

IBD100 – GATILHOS - O MODO DE IMPLEMENTAÇÃO DO POSTGRESQL

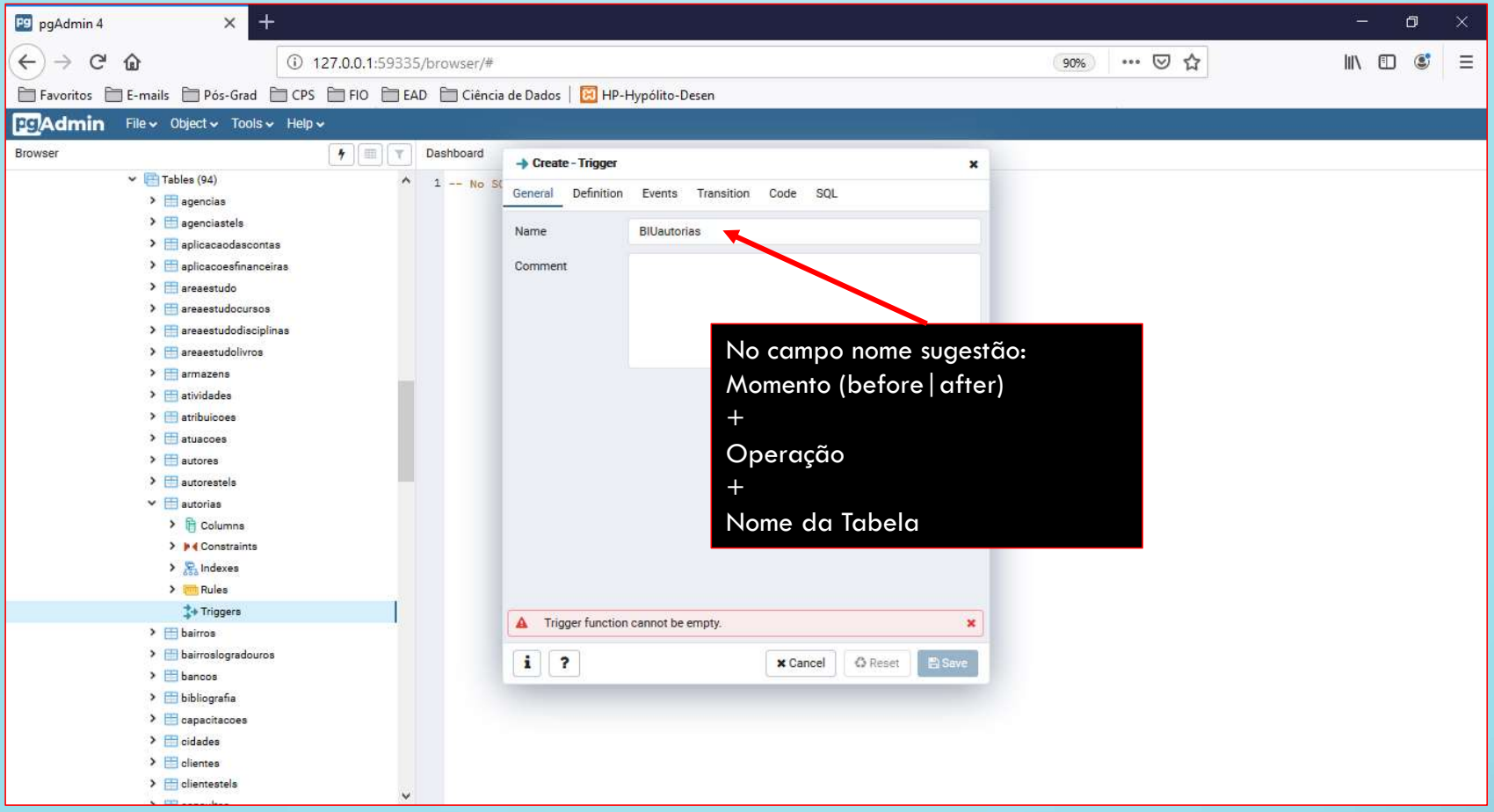
- Construindo no pgAdmin4.
- Agora, vincula-se a Trigger Function a uma tabela.
- Será disparada quando (B ou A) o evento (I,D ou U) acontecer → sugestão de nome: BA|IDU+Tabela.
- Começando...



João Maurício Hypólito - prof.joao.hypolito@gmail.com

IBD100 – GATILHOS - O MODO DE IMPLEMENTAÇÃO DO POSTGRESQL

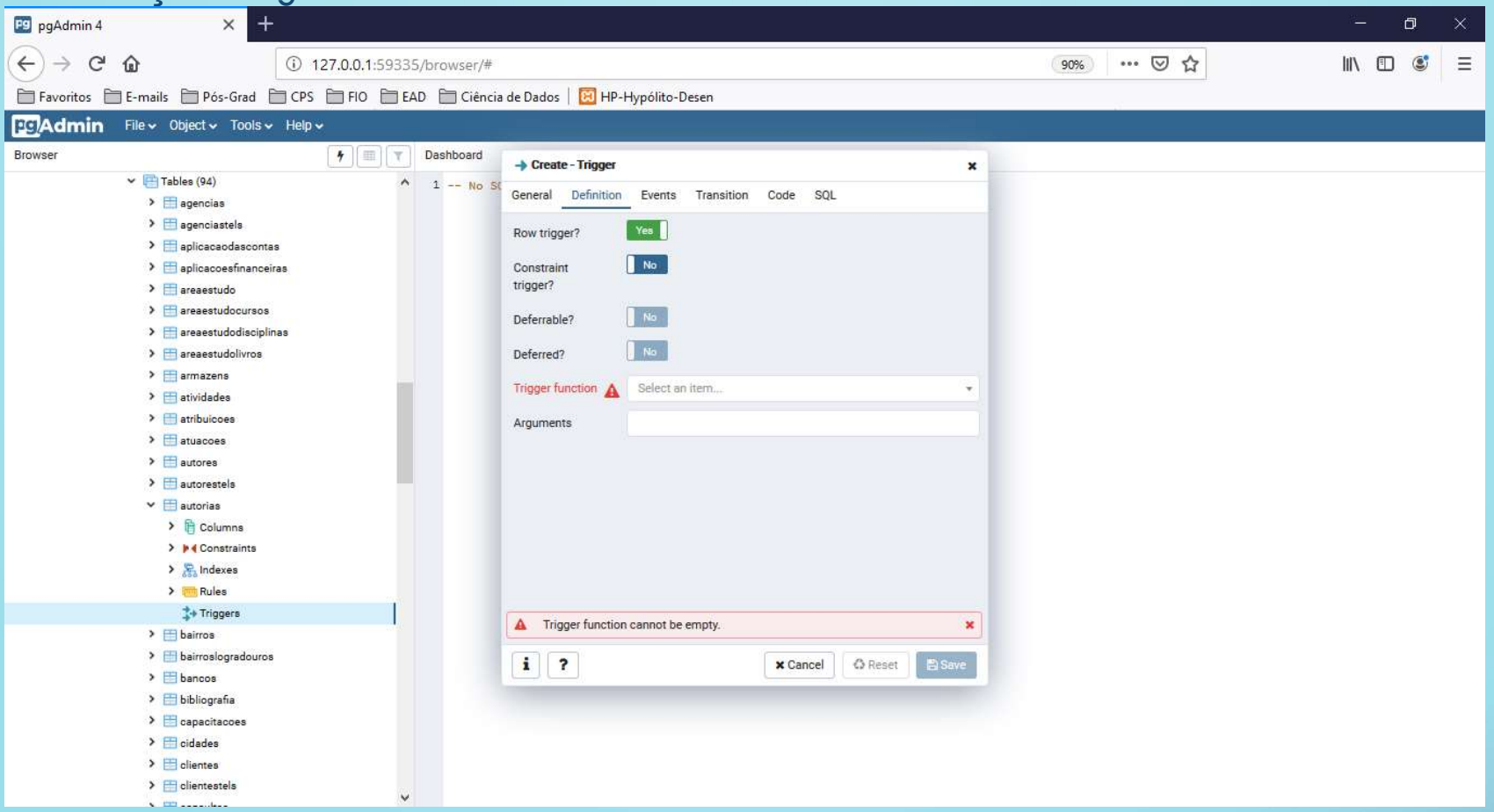
- Construindo no pgAdmin4.



João Maurício Hypólito - prof.joao.hypolito@gmail.com

IBD100 – GATILHOS - O MODO DE IMPLEMENTAÇÃO DO POSTGRESQL

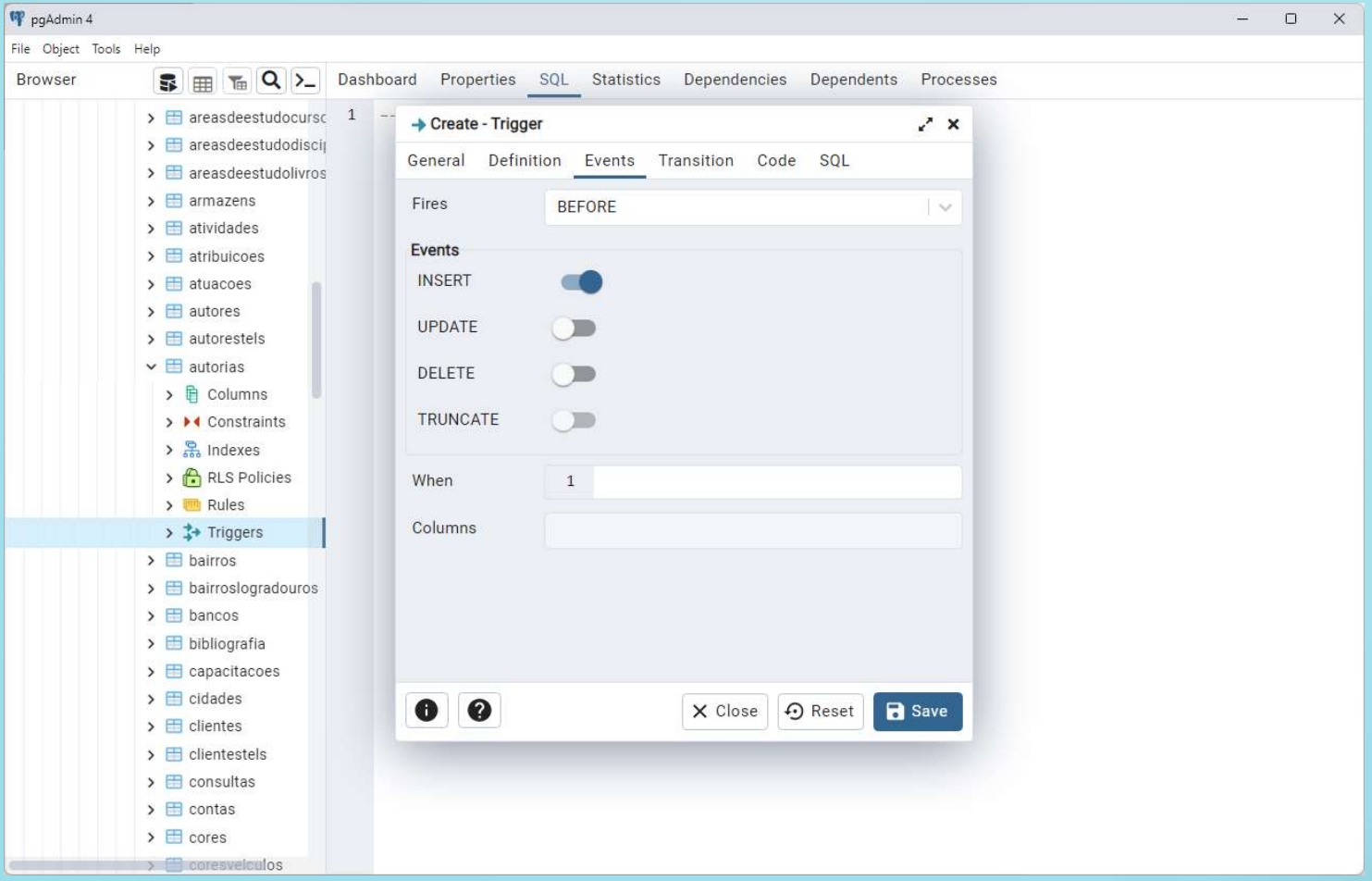
- Construindo no pgAdmin4.
- Qual função de gatilho será executada?



João Maurício Hypólito - prof.joao.hypolito@gmail.com

IBD100 – GATILHOS - O MODO DE IMPLEMENTAÇÃO DO POSTGRESQL

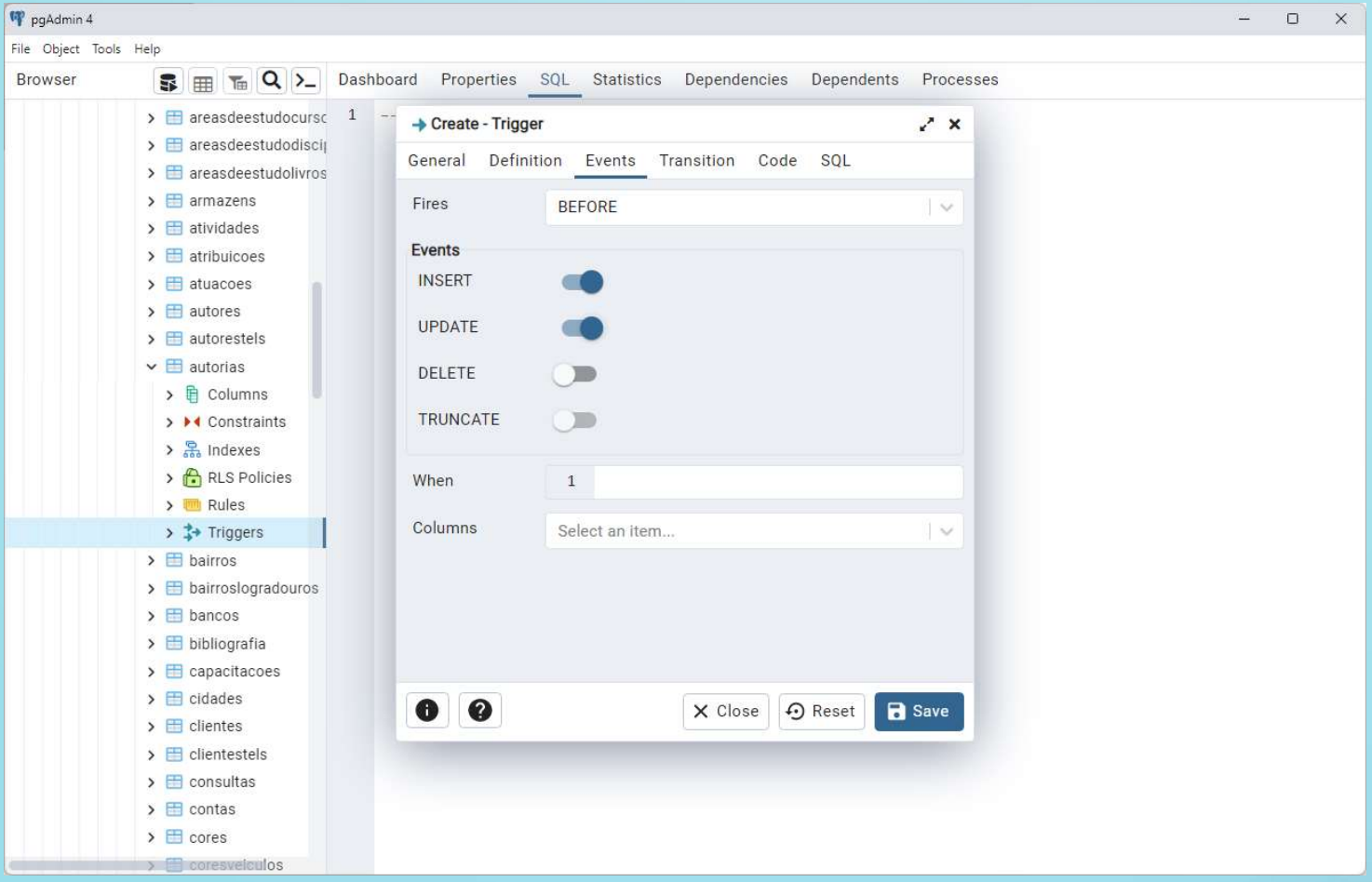
- Construindo no pgAdmin4.
- Em que momento será disparada a Trigger Function?



João Maurício Hypólito - prof.joao.hypolito@gmail.com

IBD100 – GATILHOS - O MODO DE IMPLEMENTAÇÃO DO POSTGRESQL

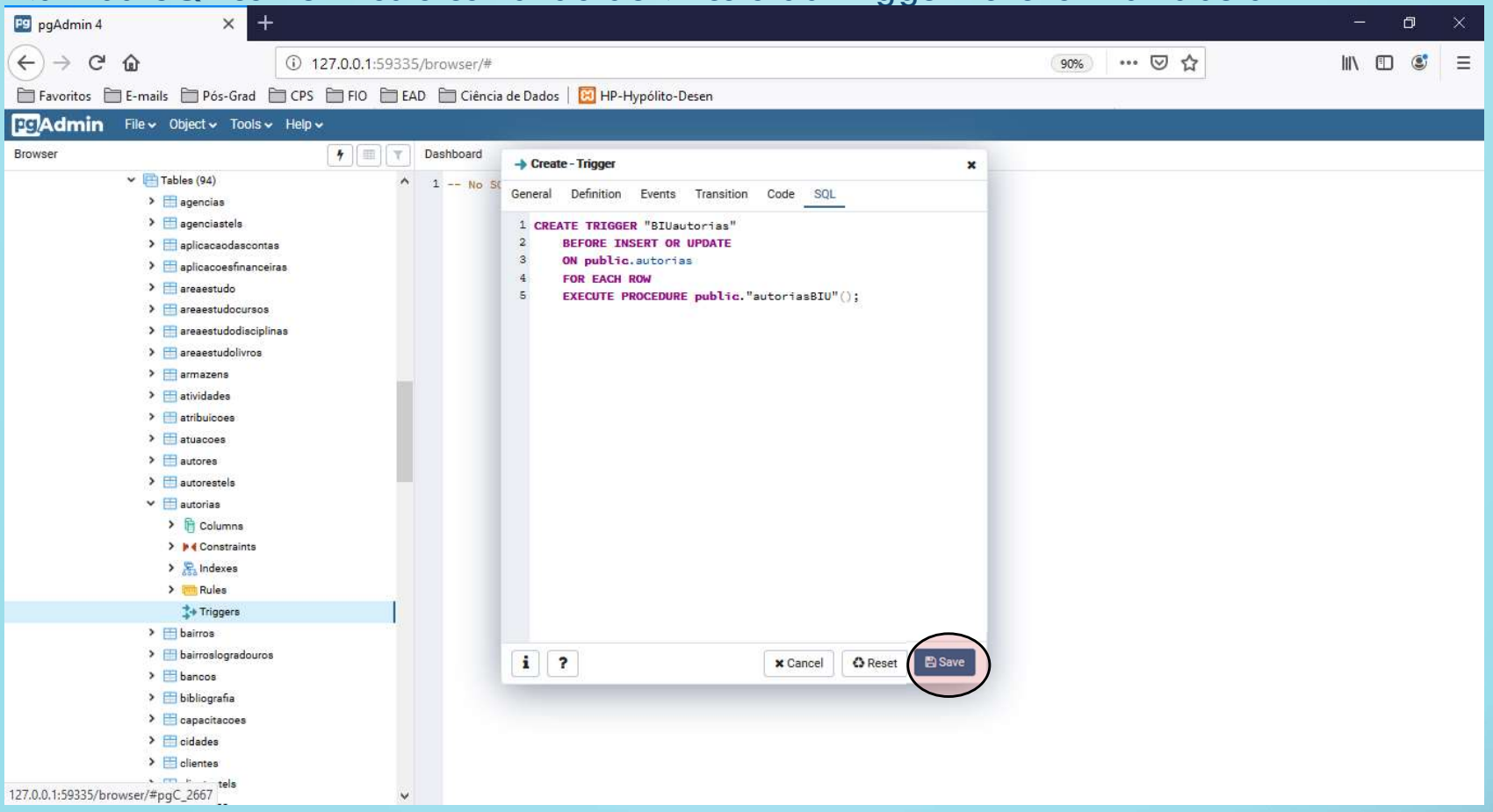
- Construindo no pgAdmin4.
- No PostgreSQL é possível disparar a mesma Trigger Function para dois ou mais eventos.



João Maurício Hypólito - prof.joao.hypolito@gmail.com

IBD100 – GATILHOS - O MODO DE IMPLEMENTAÇÃO DO POSTGRESQL

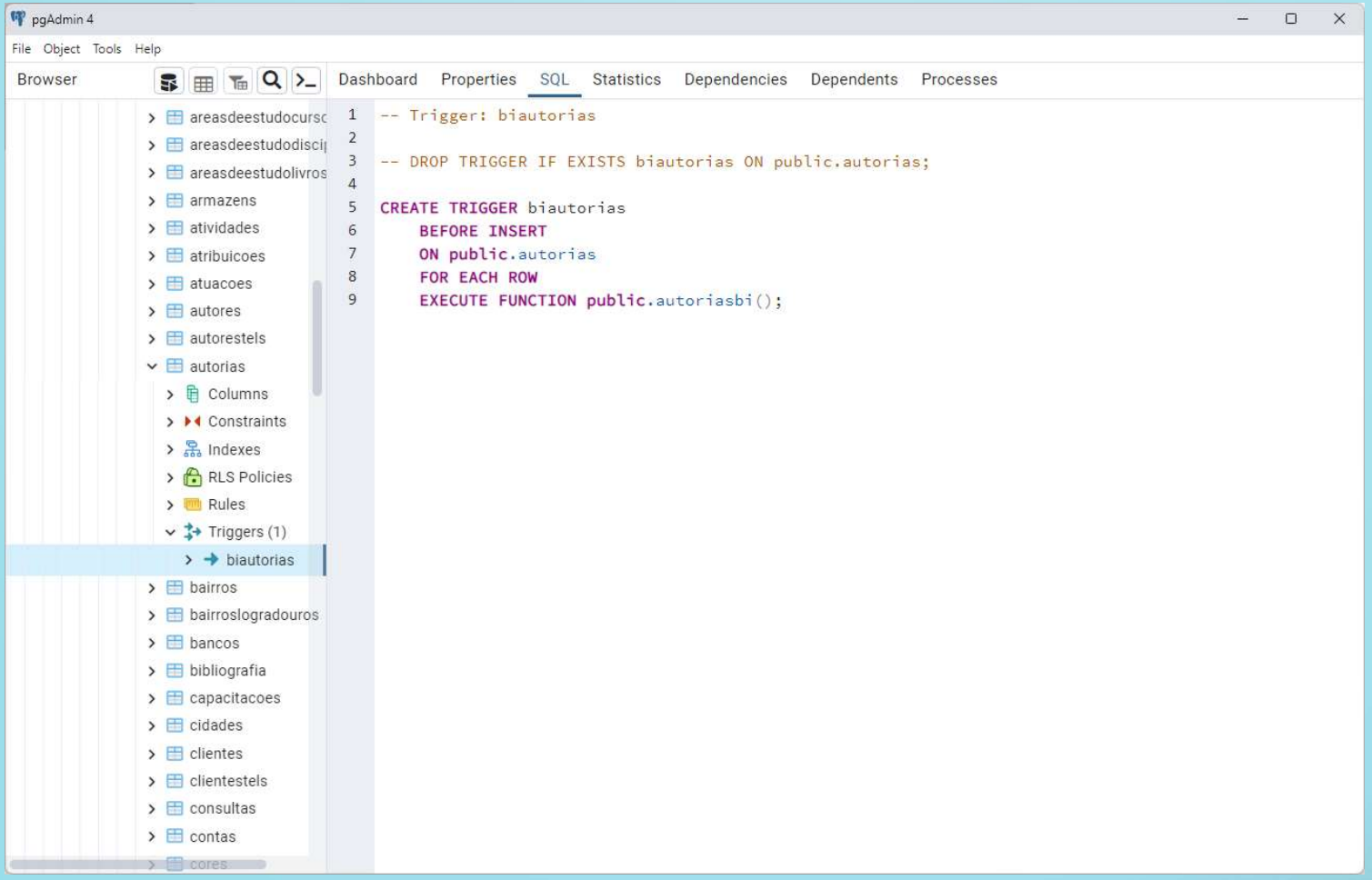
- Construindo no pgAdmin4.
- Na Aba SQL conferimos o comando de vínculo da Trigger Function na Tabela.



João Maurício Hypólito - prof.joao.hypolito@gmail.com

IBD100 – GATILHOS - O MODO DE IMPLEMENTAÇÃO DO POSTGRESQL

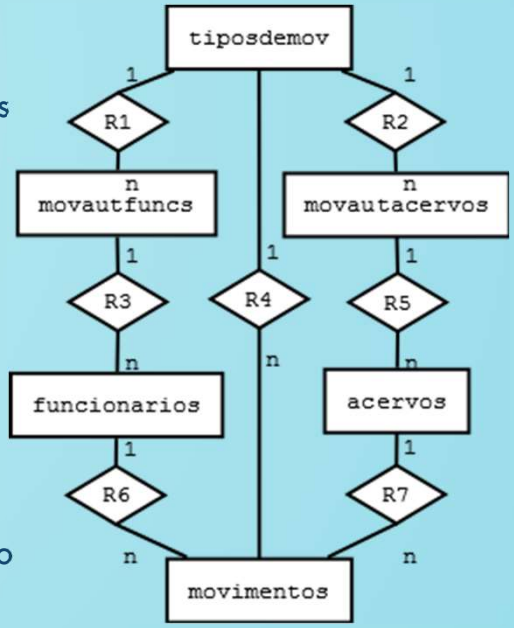
- Construindo no pgAdmin4.
- Depois de Gravar, o gatilho se apresenta assim:



João Maurício Hypólito - prof.joao.hypolito@gmail.com

IBD100 – GATILHOS - O MODO DE IMPLEMENTAÇÃO DO POSTGRESQL

- Agora vamos para algo mais complicado...
- Quando um funcionário movimentar uma peça do acervo, deve-se verificar SE o tipo de movimento pretendido (no insert movimentos) está na lista de tipos de movimentos autorizados para o funcionário E para a peça do acervo.
- Note: ANTES de fazer um insert na tabela movimentos, devemos fazer duas seleções nas tabelas movautfuncs e movautacervos e verificar se o valor do código do tipo de movimento (cdtipomov) está nas duas tabelas.
- Podemos contar a quantidade de linhas e verificar se o valor é maior que 0 (zero) nas duas contagens.
- Escreva os comandos de definição da Trigger Function e o vinculo com a tabela em um arquivo TXT e anexe na tarefa proposta no Teams



- Agora escolha **dois** dos gatilhos propostos no exercício de Gatilhos&Relatórios.
- Determine o nome do gatilho, o evento e o momento associado à execução do gatilho.
- Use o pgAdmin4 para montar a função de gatilho e o gatilho. Bom trabalho!


LABORATÓRIO DE
BANCO DE DADOS
IBD100