Vitor Alexandre

Página Inicial Conteúdo Conclusão

VERIFICADOR DE DEPENDÊNCIAS

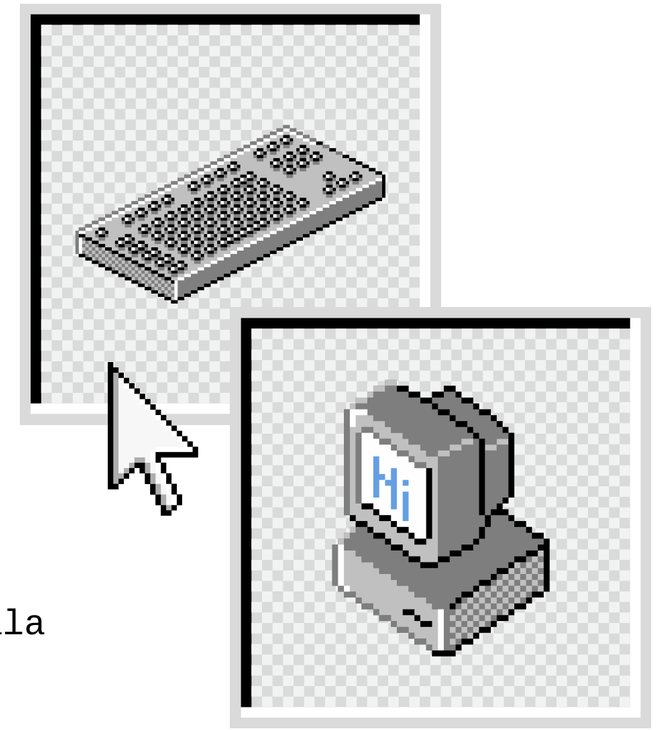
Iniciar

Criar um programa capaz de identificar automaticamente dependências funcionais em tabelas de um banco de dados, utilizando comandos SQL e lógica de programação.

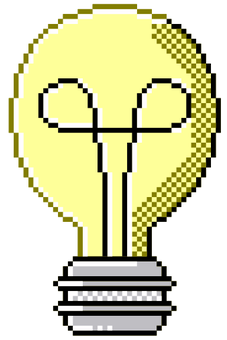
O QUE É UMA DEPENDÊNCIA FUNCIONAL?

Dizemos que A → B é uma dependência funcional se,
para cada valor de A, existe apenas um valor possível de B.

- RA → Nome
- CPF → Cliente
- Curso, Período → Sala



COMO FUNCIONA



- Lê todas as colunas da tabela informada;
- Gera combinações de 1 a 3 atributos para o lado esquerdo (X);
- Para cada X → Y, executa a query SQL:
- Se a query não retorna linhas → dependência válida.
- O sistema mostra as dependências encontradas e o tempo total.

```
SELECT 1
FROM tabela
GROUP BY X
HAVING COUNT(DISTINCT Y) > 1;
```

DOCUMENTAÇÃO

README.md → Identificação

conexaocombd.js→ Conectar com o Banco de Dados

createexemplos.csv → Criação de um Banco de Dados para Exemplos

insertexemplo.csv → Insere informações no Banco de Dados para Exemplos

verificador.js → Exemplo de Verificação sob o Banco de Dados

querymanual.sql→ Solicitação de Informações manual (Exemplo)

scriptexemplo.js → Verificador simples de dependências

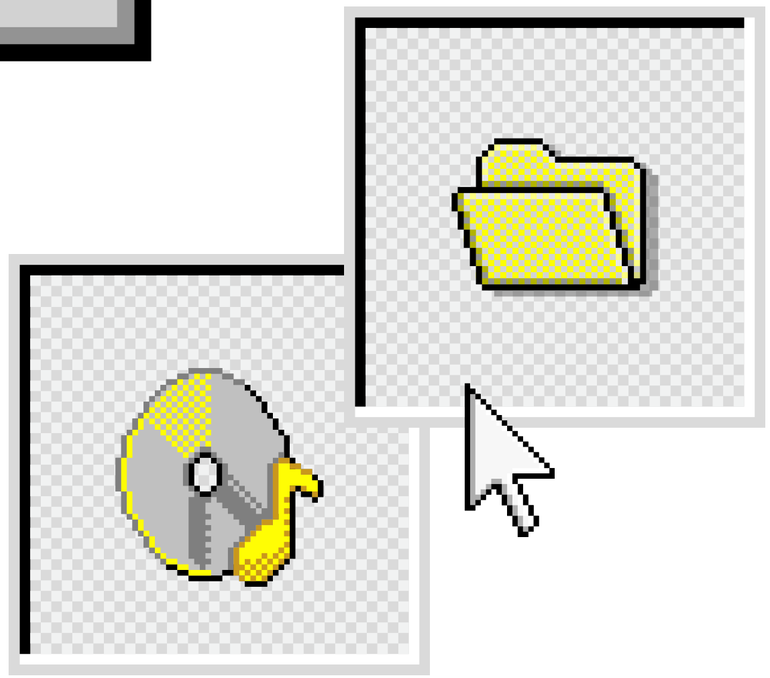
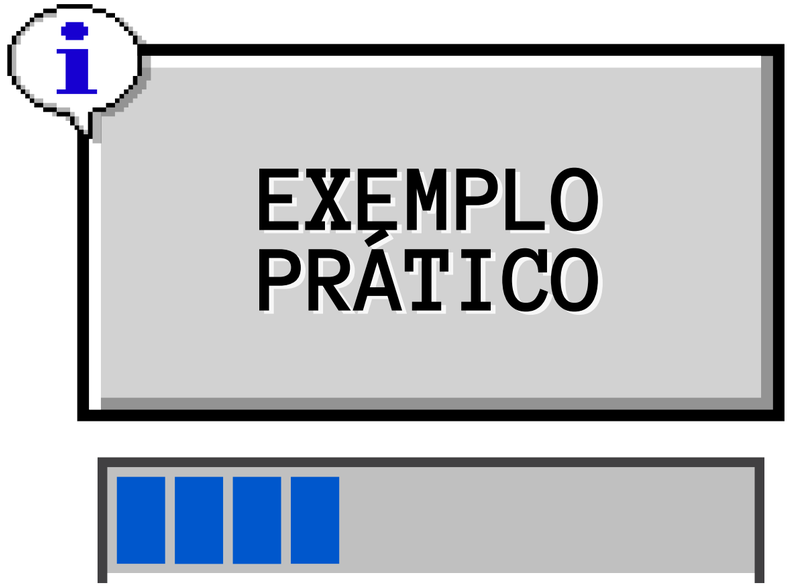


Tabela: alunos (ra, nome, curso, periodo)

- Conecta ao banco de dados PostgreSQL;
- Lê todas as tabelas do schema public;
- Gera e testa todas as combinações de colunas (até 3 atributos);
- Mostra no console as dependências funcionais válidas e o tempo total.

node verificador.js



Execução em Tabela “alunos”

- ra → nome
- ra → curso
- curso, periodo → ra

Tempo Levado: 0.42 segundos
Testes realizados: 22

ETAPAS DO PROJETO

1)Criação de uma tabela simples de teste;
2)Escrita manual de uma query SQL para testar dependências;
3)Conexão Node.js ↔ PostgreSQL com o pacote pg;
4)Implementação de função de teste de dependência;
5)Geração automática de todas as combinações possíveis;
6)Teste e validação com tabelas maiores;
7)Relatório com resultados, análise e tempo de execução.

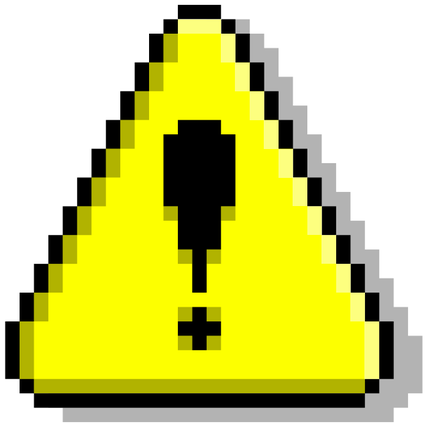
COMPLEXIDADE E LIMITAÇÕES DO PROJETO

Complexidadde:

- Se a tabela tem n colunas, o número de combinações possíveis cresce.
- Em outras palavras, quanto maior a tabela, mais tempo vai ser levado para realizar a verificação.

Limitações:

- Complexidade alta para tabelas grandes;
- O desempenho pode ser otimizado se usarmos outras formas de filtros, apesar que o atual já funcione.



OBRIGADO!

Agradeço a Atenção!

Banco de Dados | 4 Bimestre | UTFPR