

AULA 9 – SQL: DDL E DCL

PROFA. DRA. LEILA BERGAMASCO

CC5232 – Banco de Dados

NA AULA DE HOJE


- PostgreSQL
- DDL
- DCL



SGDB POSTGRESQL

- História
 - 1973 IBM lança o sistema R, primeira versão do SQL
 - Dois alunos de Berkeley se interessam e desenvolvem o próprio sistema relacional
 - Ingres
 - Suporte a tipos de dados mais complexos: Geográficos
 - Após algumas tentativas fracassadas de comercializar o Ingres, o projeto foi reformulado
 - Post-Ingres -> Postgres
 - Ganhador do Prêmio Turing!

MILESTONES

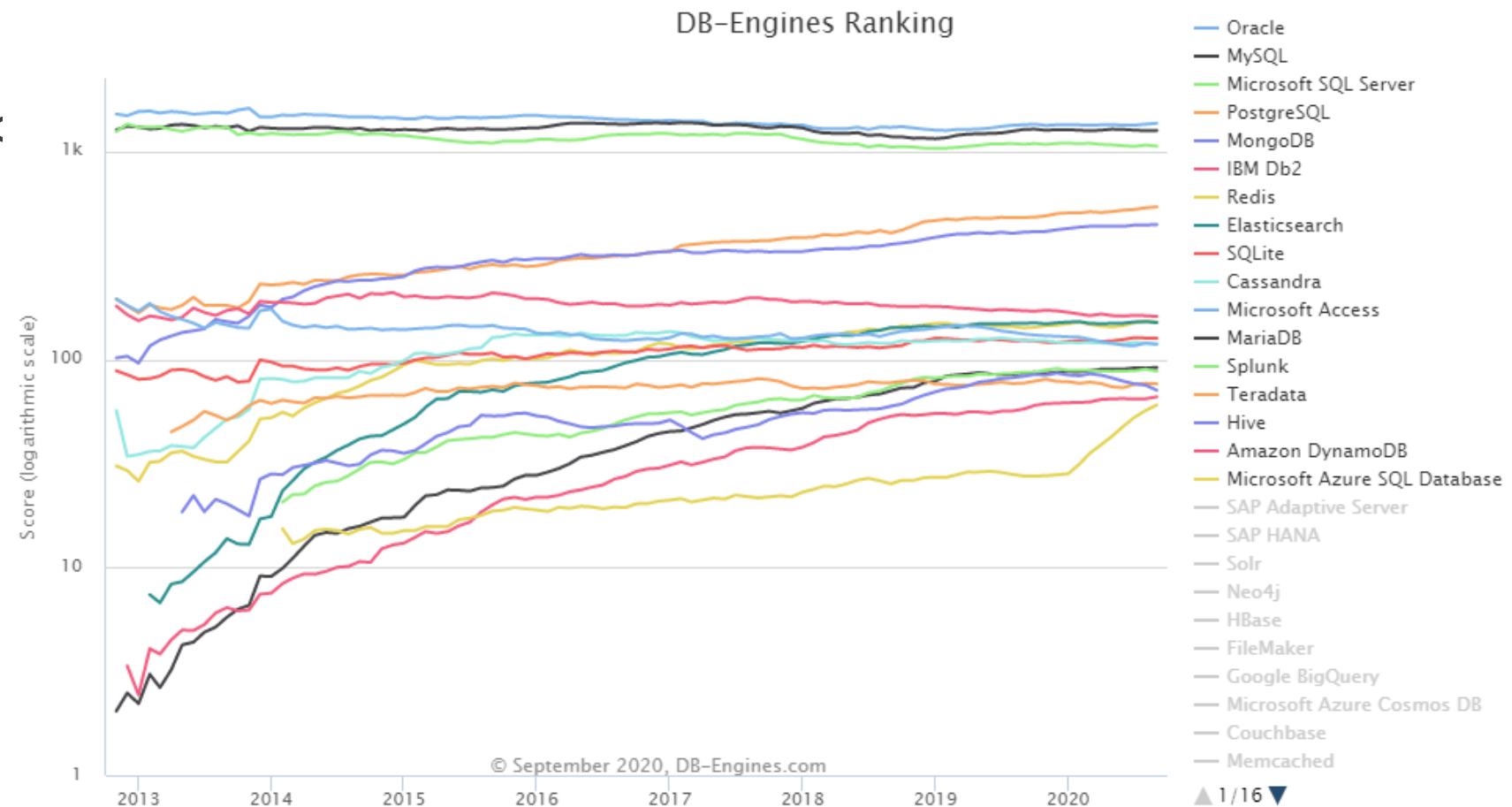
- 1985 – Postgres
- 1998 – Suporte a SQL ANSI, views, cláusulas CASE, EXCEPT, MVCC (Controle de concorrência múltipla)
 - PostgreSQL
- 2000 – join, chave estrangeira
- 2005 – compatibilidade com Windows
- 2008 – suporte ao tipo XML
- 2013 – suporte ao tipo JSON
- 2017 – diversas melhorias em performance (paralelismo, join eficiente)
- 2019 – suporte a leitura otimizada de campos JSON 

DIFERENCIAIS POSTGRESQL

- Licença BSD (Berkeley Software Distribution)
 - Considerada domínio público e pode ser modificado sem restrições
- Robusto
- Considerado SGBDOR (objeto-relacional)
- Recursos avançados
 - Explain, analyze
- Tamanho ilimitado de linhas
- Lock para controle de concorrência
- Tipos de dados especializados
 - ex. SIG - Sistemas de Informações Geográficas

DIFERENCIAIS POSTGRESQL

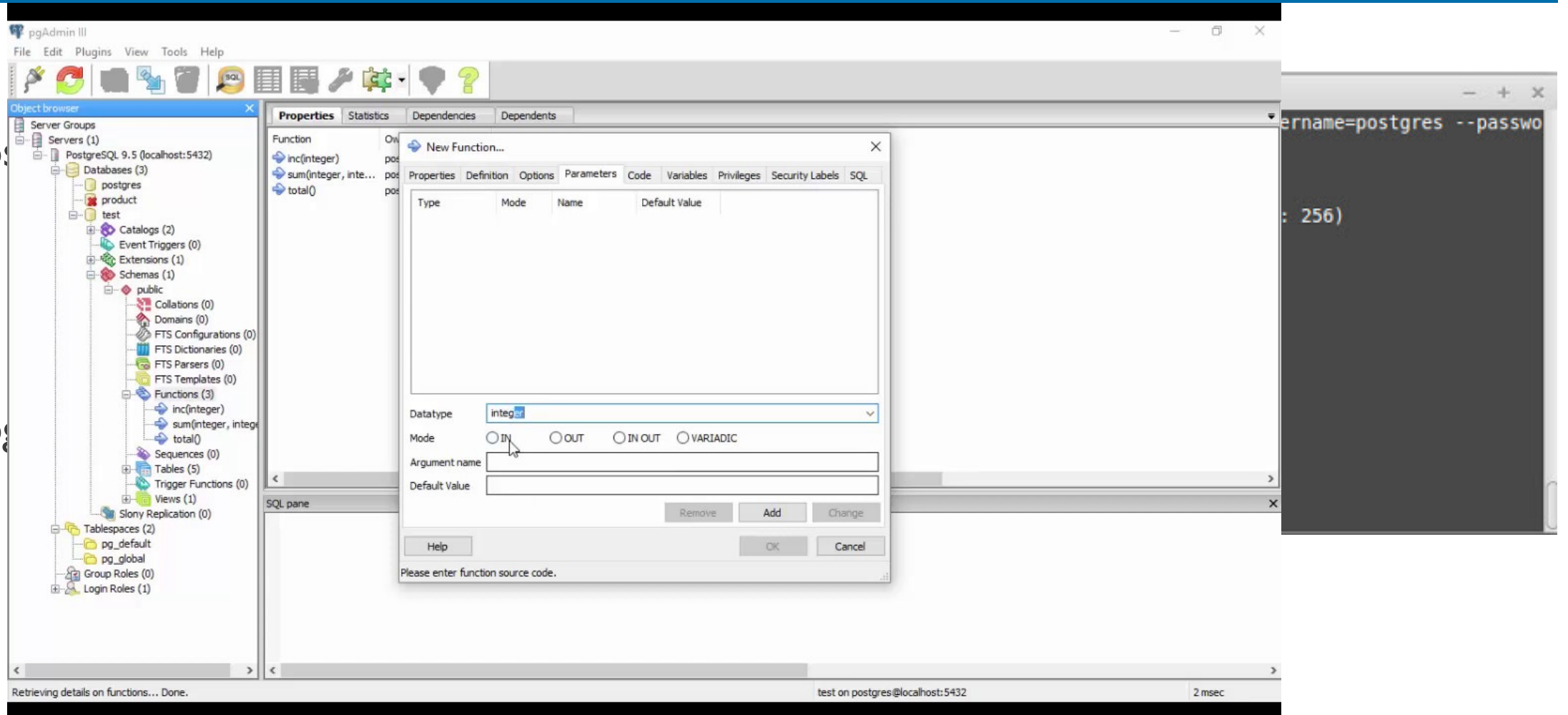
- Mantido via doações e
 - RedHat
 - Fujitsu
 - Skype



ARQUITETURA POSTGRESQL

- Cliente – Servidor
 - 2 até n-camadas: em geral 2
 - Uma aplicação *frontend* do usuário (ex: programa psql/PgAdmin) e
 - Um ou mais servidores *backend* do próprio banco de dados PostgreSQL
- Nosso exemplo
 - PgAdmin: Cliente
 - ElephantSQL: Servidor

FERRAMENTAS POSTGRESQL



HISTÓRIA DO SQL

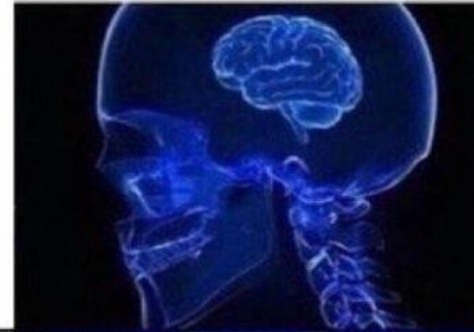
- O SQL foi desenvolvido originalmente no início dos anos 70 nos laboratórios da IBM em San Jose, dentro do projeto System R, que tinha por objetivo demonstrar a viabilidade da implementação do modelo relacional proposto por E. F. Codd.
- O nome original da linguagem era SEQUEL, acrônimo para "Structured English Query Language" (Linguagem de Consulta Estruturada, em Inglês).

HISTÓRIA DO SQL

- Padrão de banco de dados.
 - Simplicidade e facilidade de uso.
 - Especifica a forma do resultado e não o caminho para chegar a ele → descritiva
- Embora o SQL tenha sido originalmente criado pela IBM, rapidamente surgiram várias linguagens muito parecidas em outros sistemas proprietários.
- Essa expansão levou à necessidade de ser criado e adaptado um padrão para a linguagem.
 - American National Standards Institute (ANSI) em 1986 e ISO em 1987.
- O SQL foi revisto em 1992 e a esta versão foi dado o nome de SQL-92.
- Revisto novamente em 1999 e 2003 para se tornar SQL:1999 (SQL3) e SQL:2003, respectivamente.
 - O SQL:1999 usa expressões regulares de emparelhamento, queries recursivas e gatilhos (triggers).
 - O SQL:2003 introduz características relacionadas ao XML

SQL

“Structured
Query
Language”



“S-Q-L”



“Se-quel”



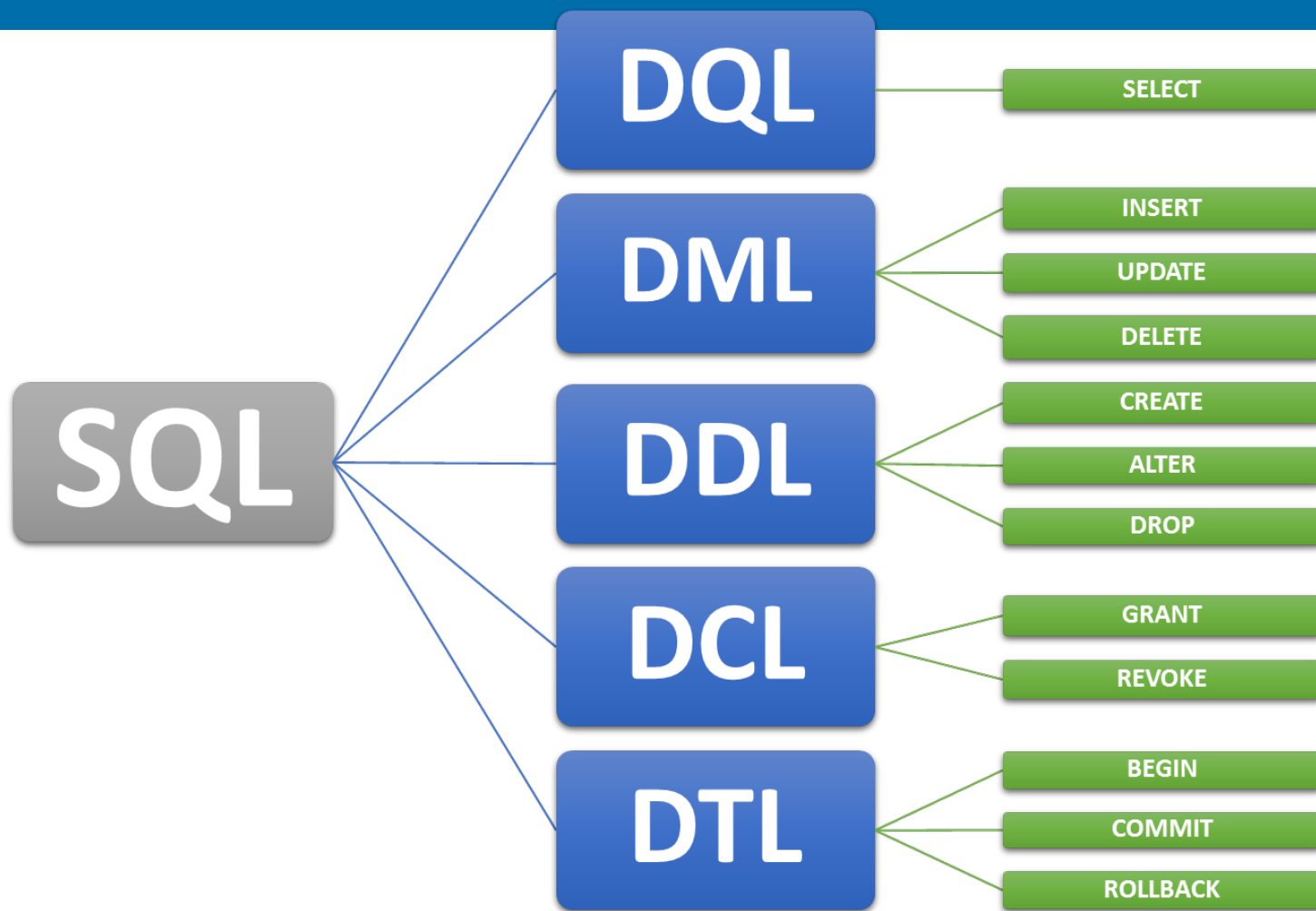
“Skewl”
“Squeal”
“Squiggle”



ORGANIZAÇÃO DO SQL

- **DDL - Linguagem de Definição de Dados** - Define os comandos utilizados para criação (CREATE) de tabelas, views, índices, atualização dessas estruturas (ALTER), assim como a remoção (DROP);
- **DCL - Linguagem de Controle de Dados** - Define os comandos utilizados para controlar o acesso aos dados do banco, adicionando (GRANT) e removendo (REVOKE) permissões de acesso;
- **DQL - Linguagem de Consulta de Dados** - Define o comando utilizado para que possamos consultar (SELECT) os dados armazenados no banco;
- **DML - Linguagem de Manipulação de Dados** - Define os comandos utilizados para manipulação de dados no banco (INSERT, UPDATE e DELETE);
- **DTL - Linguagem de Transação de Dados** - Define os comandos utilizados para gerenciar as transações executadas no banco de dados, como iniciar (BEGIN) uma transação, confirmá-la (COMMIT) ou desfazê-la (ROLLBACK).

ORGANIZAÇÃO DO SQL



DDL

- Define a estrutura dos dados e tabelas. Os comandos DDL mais comuns são **CREATE, ALTER, DROP, RENAME e TRUNCATE.**

- CREATE: database, tabela, usuário

```
CREATE DATABASE bd
```

```
CREATE SCHEMA cc0900
```

```
CREATE TABLE cc0900.aula
(
  nome VARCHAR(255),
  qtde_alum INT,
  codigo_disc VARCHAR(80),
  primary key (codigo_disc)
)
```

```
CREATE TABLE cc5232.aula2022
(
  nome VARCHAR(255),
  qtde_alum int,
  codigo_disc int,
  FOREIGN KEY (codigo_disc) REFERENCES
  cc5232.disciplina(disc_codigo)
)
```

```
CREATE USER userA WITH PASSWORD 'test123'
```

SELECT * FROM

Select * From

select * from

SeLEct * fRoM

SQL syntax



DDL

- Define a estrutura dos dados e tabelas. Os comandos DDL mais comuns são **CREATE, ALTER, DROP, RENAME e TRUNCATE**.
- ALTER: adiciona ou remove colunas de uma tabela

```
ALTER TABLE cc0900.aula ADD prof_nome VARCHAR(1000)
```

```
ALTER TABLE cc0900.aula drop column prof_nome
```

DDL

- Define a estrutura dos dados e tabelas. Os comandos DDL mais comuns são **CREATE, ALTER, DROP, RENAME e TRUNCATE**.
 - DROP: remove tabelas, databases e indices

```
ALTER TABLE cc0900.aula DROP INDEX aula_pkey
```

```
DROP TABLE cc0900.aula
```

```
DROP DATABASE bd
```


DDL

- Define a estrutura dos dados e tabelas. Os comandos DDL mais comuns são **CREATE, ALTER, DROP, RENAME e TRUNCATE**.
 - RENAME: Comando utilizado para renomear tabelas

```
ALTER TABLE cc0900.aula RENAME TO aulas
```

DCL

- São usados para controle de acesso e gerenciamento de permissões para usuários em no banco de dados. Com eles, pode-se facilmente permitir ou negar algumas ações para usuários nas tabelas ou registros (segurança de nível de linha). Os comandos mais utilizados são GRANT e REVOKE
- GRANT: atribui privilégios de acesso do usuário a objetos do banco de dados. A nível de database, esquema e ou/tabelas. Além disso permite escolher se o usuário poderá realizar SELECT, INSERT, DELETE.

```
GRANT SELECT, INSERT, UPDATE, DELETE ON ALL TABLES IN SCHEMA cc0900  
TO userA;
```

```
GRANT SELECT ON ALL TABLES IN SCHEMA cc0900 TO userA;
```

DCL

- São usados para controle de acesso e gerenciamento de permissões para usuários em no banco de dados. Com eles, pode-se facilmente permitir ou negar algumas ações para usuários nas tabelas ou registros (segurança de nível de linha). Os comandos mais utilizados são GRANT, REVOKE e DENY
 - REVOKE: Remove os privilégios de acesso aos objetos obtidos com o comando GRANT

```
REVOKE ALL PRIVILEGES ON cc0900 FROM userA
```

SETUP ELEPHANTSQL E PGADMIN

- Seção Procedimentos no Moodle
- Utilizaremos o plano “Free” disponível.
- Necessário somente uma conta de e-mail ativa (por aluno).
- Performance e instância compartilhada do serviço (Free).
- Tamanho máximo da base de dados 20 MB.
 - Suficiente para os exercícios das aulas.

EXERCÍCIOS

Considere os seguintes esquemas de relação:

- Professor (prof-numero, prof-nome, prof-rua, prof-cidade, prof-telefone)
- Aluno (alu-numero, alu-nome, alu-rua, alu-cidade)
- Disciplina (disc-codigo, disc-nome, disc-quant-aulas-semana)
- Matricula(**alu-numero**, **disc-codigo**, ano, semestre, nota, frequencia)
- ProfessorDisciplina (**prof-numero**, **disc-codigo**)

- Escreva a consulta SQL considerando os comandos DDL. Campos sublinhados = chave primária e campos azuis chaves estrangeiras.

OBRIGADO E ATÉ A PRÓXIMA AULA!