

BANCO DE DADOS I

The background features a large, dark blue arrow pointing to the right, which is partially overlaid by a light blue shape. Below the arrow, there is a horizontal orange bar. The overall design is modern and minimalist.



EMENTA

- Modelagem de Dados: modelos conceituais e modelos relacional.
- Sistemas de Gerenciamento de Banco de Dados.
- Normalização.
- Abordagem relacional semântica, modelo externo, álgebra relacional e cálculo relacional.



EMENTA

- Arquitetura de um sistema gerenciador de banco de dados.
- Projeto de banco de dados: conceitual, lógico e físico.
- Linguagem SQL (DDL, DML), triggers e procedures.



CARGA HORÁRIA

- TOTAL: 60H
- QUANTIDADE DE AULAS: 80



AVALIAÇÃO

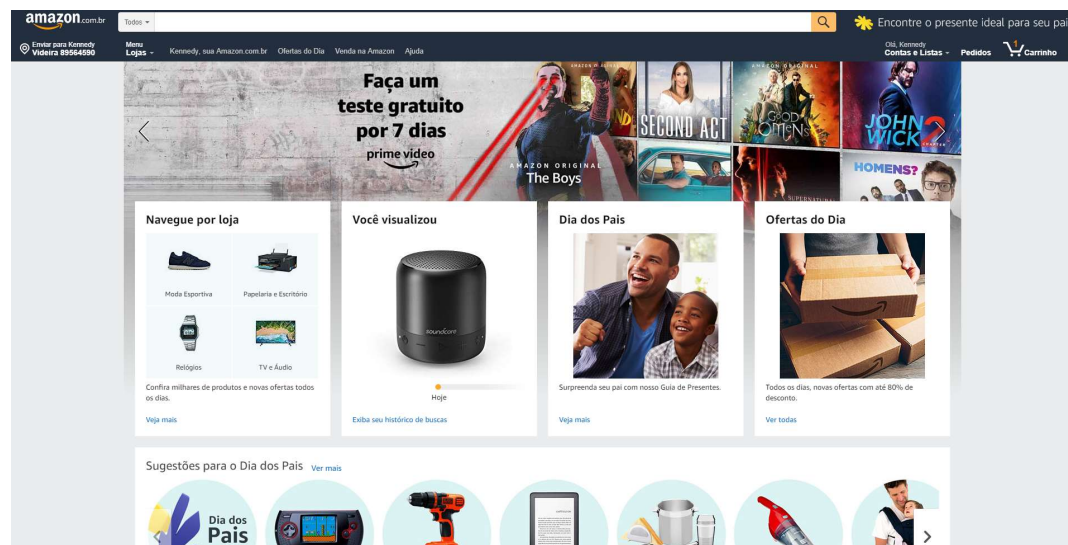
- PROVA – PESO 4
- PROJETO – PESO 4
- ATIVIDADES – PESO 2


INTRODUÇÃO

Uma grande jornada começa com o primeiro passo...



UM SISTEMA PRECISA DE DADOS





NO PRINCÍPIO
ERAM FICHAS QUE SE
TORNARAM
ARQUIVOS DIGITAIS
ATÉ QUE SE OUVIU:
PRECISAMOS
OTIMIZAR NOSSO
TRABALHO...



EVOLUÇÃO

- 60, com apoio da IBM início de pesquisas na área
- 60, modelos hierárquico e de rede
- 70, modelo relacional
- Final de 70, início de 80 Sistemas de Gerenciamento de Banco de Dados Relacionais (SGBDR)



EVOLUÇÃO

- Inicialmente lento, mas a posteriori conseguiram melhor desempenho
- 80, Banco de dados orientado a objeto
- 80, BD00s – complexo e despadronizado



UM SISTEMA RELACIONAL É AQUELE QUE:

- Os dados são percebidos pelo usuário como tabelas (e nada além de tabelas).
- Os operadores à disposição do usuário são aqueles que geram novas tabelas a partir de tabelas antigas.



ESTRUTURA GERAL

Como é o esqueleto?

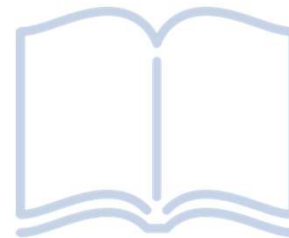




TABELA EXEMPLO

id_empregado	nome	email	salario	setor	funcao
1	João Silva	joao@gmail.com	1500	financeiro	gerente
2	Julia Alves	julia@gmail.com	2000	administrativo	analista
3	Lucas	lucas@gmail.com	1800	financeiro	analista financeiro
4	Marcos	marcos@gmail.com	3000	serviços gerais	atendente
5	Nicolas	nicolas@gmail.com	1700	administrativo	analista
6	Luiz	luiz@gmail.com	1750	serviços gerais	atendente



TABELA EXEMPLO

empregado
id_employado INT
nome VARCHAR(45)
email VARCHAR(45)
salario DOUBLE
setor VARCHAR(45)
funcao VARCHAR(45)
Indexes

Result Grid

Filter Rows:

Edit:

Export/Import:

	id_employado	nome	email	salario	setor	funcao
▶	1	João Silva	joao@gmail.com	1500	financeiro	gerencia
	2	Julia Alves	julia@gmail.com	2000	administrativo	analista
	3	Lucas	lucas@gmail.com	1800	financeiro	analista financeiro
	4	Marcos	marcos@gmail.com	3000	serviços gerais	atendente
	5	Nicolas	nicolas@gmail.com	1700	administrativo	analista
	6	Luiz	luiz@gmail.com	1750	serviços gerais	atendente
✱	NULL	NULL	NULL	NULL	NULL	NULL



TIPO DE DADOS

número

data

string



TIPO DE DADOS

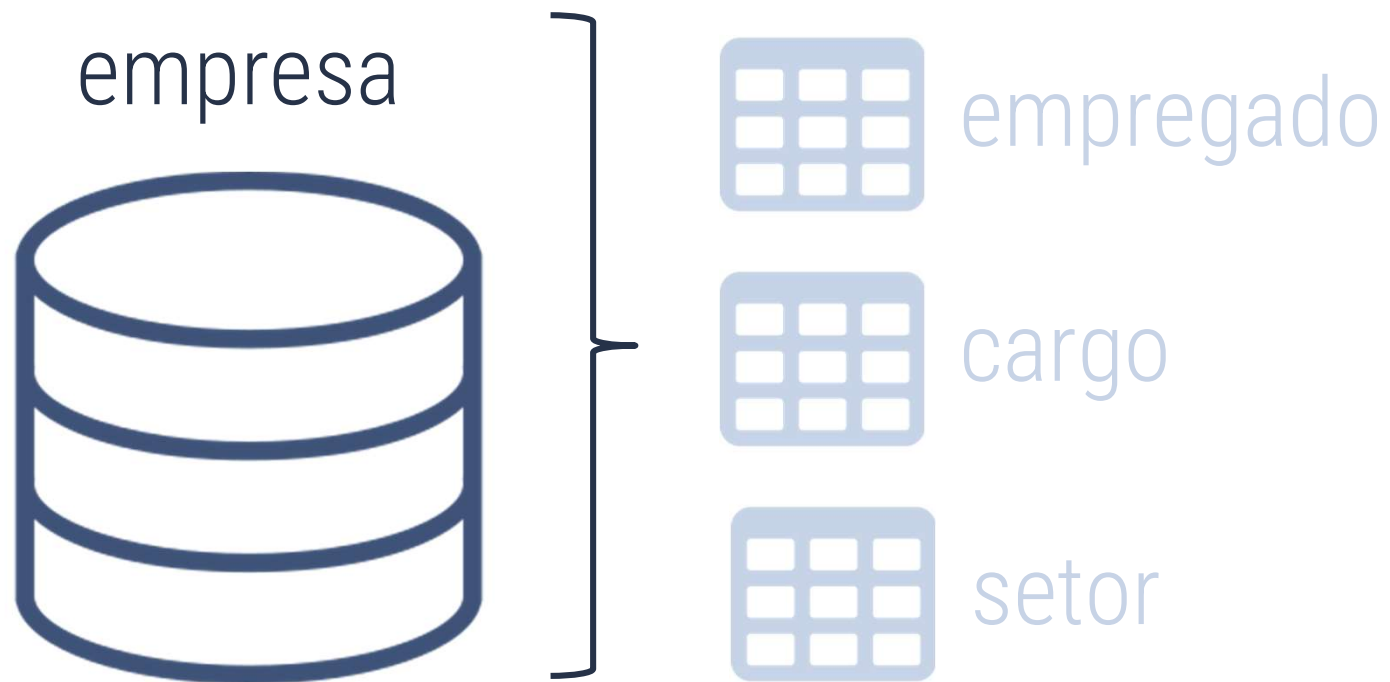
MySQL DATA TYPES

DATE TYPE	SPEC	DATA TYPE	SPEC
CHAR	String (0 - 255)	INT	Integer (-2147483648 to 2147483647)
VARCHAR	String (0 - 255)	BIGINT	Integer (-9223372036854775808 to 9223372036854775807)
TINYTEXT	String (0 - 255)	FLOAT	Decimal (precise to 23 digits)
TEXT	String (0 - 65535)	DOUBLE	Decimal (24 to 53 digits)
BLOB	String (0 - 65535)	DECIMAL	"DOUBLE" stored as string
MEDIUMTEXT	String (0 - 16777215)	DATE	YYYY-MM-DD
MEDIUMBLOB	String (0 - 16777215)	DATETIME	YYYY-MM-DD HH:MM:SS
LONGTEXT	String (0 - 4294967295)	TIMESTAMP	YYYYMMDDHHMMSS
LOBLOB	String (0 - 4294967295)	TIME	HH:MM:SS
TINYINT	Integer (-128 to 127)	ENUM	One of preset options
SMALLINT	Integer (-32768 to 32767)	SET	Selection of preset options
MEDIUMINT	Integer (-8388608 to 8388607)	BOOLEAN	TINYINT(1)

Copyright © mysqltutorial.org. All rights reserved.

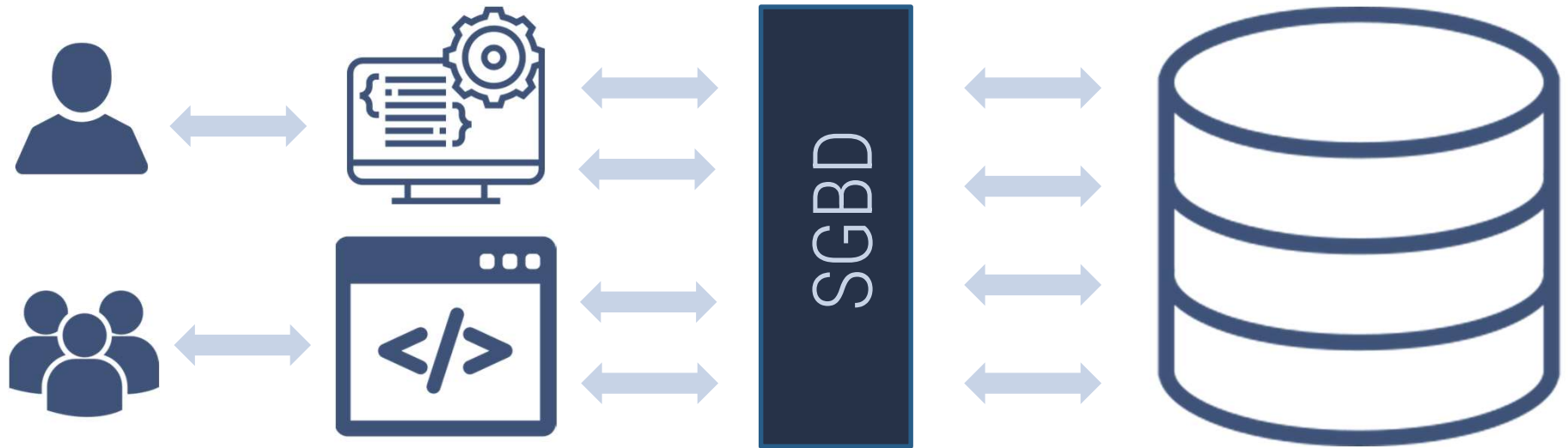


EXEMPLO






SGBD





SGBD – Mais utilizados

DB-ENGINES
Knowledge Base of Relational and NoSQL Database Management Systems

 **CrateDB**
Get the capabilities and services to make your real-time SQL projects a success.

English
[Deutsch](#)

provided by [solid.IT](#)

[Home](#) | [DB-Engines Ranking](#) | [Systems](#) | [Encyclopedia](#) | [Blog](#) | [Search](#) | [Vendor Login](#)


Featured Products: [Neo4j](#) [RavenDB](#) [Couchbase](#) [Redis](#) [DataStax](#)

Select a ranking

- Complete ranking
- Relational DBMS
- Key-value stores
- Document stores
- Graph DBMS
- Time Series DBMS
- Object oriented DBMS
- RDF stores
- Search engines
- Wide column stores
- Multivalued DBMS
- Native XML DBMS
- Event Stores
- Content stores
- Navigational DBMS

Special reports

- Ranking by database model
- Open source vs. commercial


Featured Products
 **redislabs**
HOME OF REDIS
Start now with Redis Cloud
Server, highly available Redis


[Ranking > Complete Ranking](#)

DB-Engines Ranking

The DB-Engines Ranking ranks database management systems according to their popularity. The ranking is updated monthly.

Read more about the [method](#) of calculating the scores.

 [RSS](#) [RSS Feed](#)


trend chart

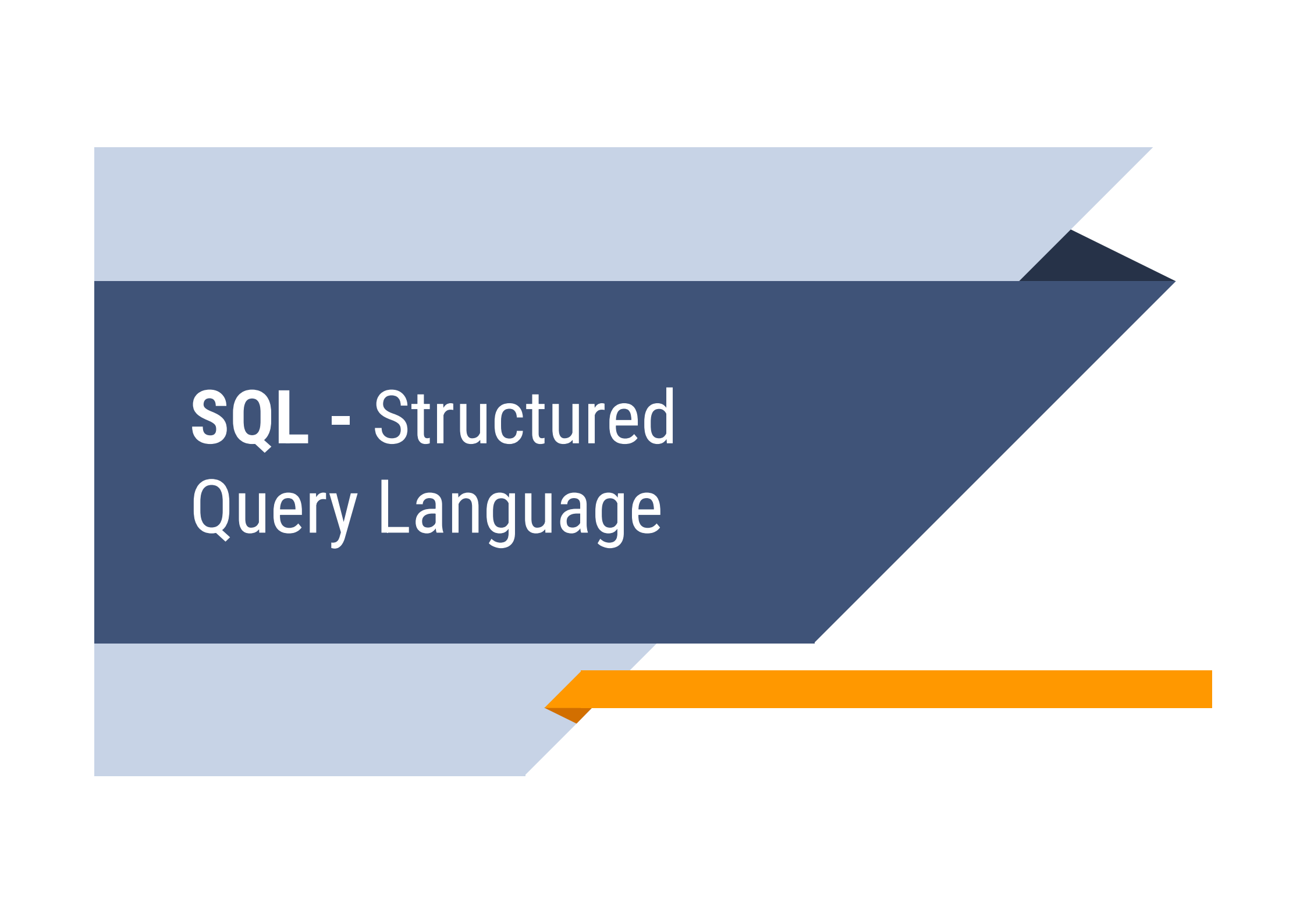
351 systems in ranking, August 2019

Rank			DBMS	Database Model	Score		
Aug 2019	Jul 2019	Aug 2018			Aug 2019	Jul 2019	Aug 2018
1.	1.	1.	Oracle	Relational, Multi-model	1339.48	+18.22	+27.45
2.	2.	2.	MySQL	Relational, Multi-model	1253.68	+24.16	+46.87
3.	3.	3.	Microsoft SQL Server	Relational, Multi-model	1093.18	+2.35	+20.53
4.	4.	4.	PostgreSQL	Relational, Multi-model	481.33	-1.94	+63.83
5.	5.	5.	MongoDB	Document	404.57	+5.36	+53.59
6.	6.	6.	IBM Db2	Relational, Multi-model	172.95	-1.19	-8.89
7.	7.	8.	Elasticsearch	Search engine, Multi-model	149.08	+0.27	+10.97
8.	8.	7.	Redis	Key-value, Multi-model	144.08	-0.18	+5.51
9.	9.	9.	Microsoft Access	Relational	135.33	-1.98	+6.24
10.	10.	10.	Cassandra	Wide column	125.21	-1.80	+5.63
11.	11.	11.	SQLite	Relational	122.72	-1.91	+8.99
12.	12.	13.	Splunk	Search engine	85.88	+0.39	+15.39
13.	13.	14.	MariaDB	Relational, Multi-model	84.95	+0.52	+16.66
14.	14.	18.	Hive	Relational	81.80	+0.93	+23.86
15.	15.	12.	Teradata	Relational, Multi-model	76.64	-1.18	-0.77



VANTAGENS

- Os dados podem ser compartilhados
- A redundância pode ser reduzida
- A inconsistência pode ser evitada (até certo ponto)
- A integridade pode ser mantida

The background features a light blue geometric shape on the left, a dark blue arrow-like shape pointing right in the center, and a horizontal orange bar at the bottom right.

SQL - Structured Query Language



CLASSIFICAÇÃO SQL

DML
Manipulação

SELECT
INSERT
UPDATE
DELETE

DTL
Transação

BEGIN WORK
COMMIT
ROLLBACK

DDL
Definição

CREATE
DROP

DCL
Controle

GRANT
REVOKE

DQL
Consulta

WHERE
GROUP BY
HAVING
ORDER BY
DISTINCT

DQL – PARTE I

Linguagem de Consulta de Dados



TABELA EXEMPLO

id_empregado	nome	email	salario	setor	funcao
1	João Silva	joao@gmail.com	1500	financeiro	gerente
2	Julia Alves	julia@gmail.com	2000	administrativo	analista
3	Lucas	lucas@gmail.com	1800	financeiro	analista financeiro
4	Marcos	marcos@gmail.com	3000	serviços gerais	atendente
5	Nicolas	nicolas@gmail.com	1700	administrativo	analista
6	Luiz	luiz@gmail.com	1750	serviços gerais	atendente



TABELA EXEMPLO

empregado	
id_employado	INT
nome	VARCHAR(45)
email	VARCHAR(45)
salario	DOUBLE
setor	VARCHAR(45)
funcao	VARCHAR(45)
Indexes	

Result Grid						
		Filter Rows:		Edit:		Export/Import:
	id_employado	nome	email	salario	setor	funcao
▶	1	João Silva	joao@gmail.com	1500	financeiro	gerencia
	2	Julia Alves	julia@gmail.com	2000	administrativo	analista
	3	Lucas	lucas@gmail.com	1800	financeiro	analista financeiro
	4	Marcos	marcos@gmail.com	3000	serviços gerais	atendente
	5	Nicolas	nicolas@gmail.com	1700	administrativo	analista
	6	Luiz	luiz@gmail.com	1750	serviços gerais	atendente
*	NULL	NULL	NULL	NULL	NULL	NULL



SELECT coluna1,coluna2 **FROM** tabela

FORMA BASE



SELECT * FROM tabela

CURINGA – TODAS AS COLUNAS



```
SELECT *  
FROM tabela as novoNome
```

ALIAS – Renomear tabela ou coluna



SELECT nome **as** funcionario
FROM tabela **as** emp

ALIAS – Renomear tabela ou coluna



SELECT nome funcionario
FROM empregado emp

ALIAS – Renomear tabela ou coluna



DISTINCT

■ Restringe a seleção a apenas valores distintos/diferentes

SELECT DISTINCT coluna **FROM** tabela




■ Ex: Listar quais os setores que estão presentes no banco

SELECT DISTINCT setor **FROM** empregado




OPERADORES ARITMÉTICOS

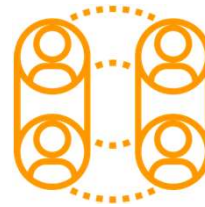
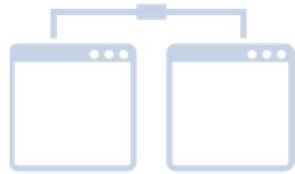
OPERADOR	DESCRIÇÃO
+	SOMA
-	SUBTRAÇÃO
/	DIVISÃO
*	MULTIPLICAÇÃO
% ou MOD	RESTO
DIV	DIVISÃO INTEIRA

- 
- Ex: Listar quanto ficaria o salario de cada empregado com uma redução de 10%

```
SELECT *, (salario-(salario*0.1)) as novoSalario  
FROM empregado
```

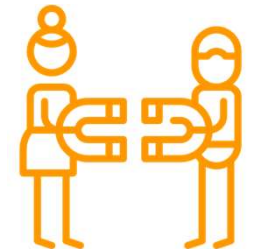
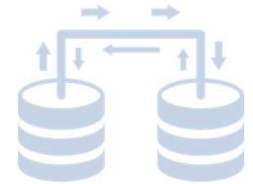
- 
- Ex: Listar quanto ficaria o salario de cada empregado com uma redução de 10%

```
SELECT *, (salario*0.9) as novoSalario  
FROM empregado
```



WHERE

Restrição com maior flexibilidade





■ Ex: Listar os empregados do setor financeiro

```
SELECT nome,setor FROM empregado  
WHERE setor='financeiro'
```



OPERADORES RELACIONAIS

OPERADOR	DESCRIÇÃO
=	IGUALDADE
<>	DIFERENTE
>	MAIOR
<	MENOR
>=	MAIOR OU IGUAL
<=	MENOR OU IGUAL




■ Ex: Listar os empregados que não sejam do setor financeiro

```
SELECT nome,setor FROM empregado  
WHERE setor<>'financeiro'
```



■ Ex: Listar os empregados que ganham mais de 2 mil reais

```
SELECT nome,setor FROM empregado  
WHERE salario>2000
```


- 
- Ex: Listar os funcionários que caso recebam um aumento de R\$300,00 terão um salário superior a R\$2.000,00

```
SELECT *  
FROM empregado  
WHERE salario+300>2000
```



OPERADORES LÓGICOS

OPERADOR	DESCRIÇÃO
AND &&	E
OR	OU
NOT !!	NÃO
XOR	OU EXCLUSIVO




■ Ex: Listar os funcionários que sejam do setor financeiro ou do setor administrativo

SELECT nome, setor,salario **FROM** empregados
WHERE setor='financeiro' **OR** setor='administrativo'

- 
- Ex: Listar os empregados que trabalhem no setor de serviços gerais e ganhem mais de R\$2.000

SELECT nome, setor,salario **FROM** empregado
WHERE setor='serviços gerais' **AND** salario>2000

- 
- Ex: Listar os empregados que ganhem entre 1500 e 2000 reais e pertençam ao setor financeiro ou administrativo

```
SELECT nome, setor,salario FROM empregado  
WHERE (salario>1500 AND salario<2000)  
AND (setor='financeiro' OR setor='administrativo')
```



PARÊNTESES

■ Com parênteses:

Result Grid			Filter Rows: <input type="text"/>	Export:	Wrap Cell Content:
	nome	setor	salario		
▶	Lucas	financeiro	1800		
	Nicolas	administrativo	1700		

■ Sem parênteses:

Result Grid			Filter Rows: <input type="text"/>	Export:	Wrap Cell Content:
	nome	setor	salario		
▶	Julia Alves	administrativo	2000		
	Lucas	financeiro	1800		
	Nicolas	administrativo	1700		

- 
- Ex: Listar os empregados ou sejam do setor financeiro ou ganhem mais de R\$2.000,00

SELECT nome, setor,salario **FROM** empregado
WHERE salario>2000 **XOR** setor='financeiro'



OPERADORES

OPERADOR	DESCRIÇÃO
BETWEEN	ENTRE
LIKE	PADRÃO TEXTUAL
IN	CONJUNTO DE VALORES



BETWEEN

- Restringe a seleção a apenas valores que se encaixem entre os valores informados (retorno inclui valor1 e valor2)

SELECT coluna **FROM** tabela
WHERE coluna **BETWEEN** valor1 **AND** valor2



LIKE

- Restringe a seleção a apenas valores que correspondam ao padrão textual fornecido

SELECT coluna **FROM** tabela
WHERE coluna **LIKE** '%texto_'

% - Qualquer quantidade de caracteres

_ - Um único caracter



■ Ex: Listar os empregados que tenham silva em seu sobrenome

```
SELECT nome,setor FROM empregado  
WHERE nome LIKE '%silva%'
```

DQL – PARTE II

Linguagem de Consulta de Dados



IN

■ Restringe a seleção a apenas valores informados dentro do in

SELECT coluna **FROM** tabela
WHERE coluna **IN** (valor1,valor2,valor3)



■ Ex: Listar os empregados que tenham o id 1,3,5

```
SELECT nome,setor FROM empregado  
WHERE id_empregado IN (1,2,3)
```

- 
- Ex: Listar os empregados do setor financeiro ou do administrativo

```
SELECT nome,setor FROM empregado  
WHERE setor IN ("financeiro","administrativo")
```



ORDER BY

■ Ordena os registros retornados

SELECT coluna **FROM** tabela
WHERE 1 ORDER BY coluna **[ASC/DESC]**



■ Ex: Listar os empregados em ordem alfabética

```
SELECT nome FROM empregado  
ORDER BY nome
```



- 
- Ex: Listar os empregados trazendo primeiro aqueles que ganham mais

```
SELECT nome FROM empregado  
ORDER BY salario DESC
```



LIMIT

■ Restringe a quantidade de registros retornados

```
SELECT coluna FROM tabela  
WHERE 1 ORDER BY coluna [ASC/DESC]  
LIMIT [2,]3
```



■ Ex: Listar os 3 empregados com menor salário da empresa

```
SELECT nome FROM empregado  
ORDER BY salario ASC  
LIMIT 3
```

- 
- Ex: Listar o 2º funcionário que tem o salário mais alto do setor financeiro

```
SELECT nome,salario FROM empregado  
WHERE setor='financeiro'  
ORDER BY salario DESC  
LIMIT 1,1
```

*LIMIT 1,1 – Depois do primeiro retorno 1 registro



SUM

■ Faz o somatório de determinada coluna

```
SELECT SUM(coluna) FROM tabela
```



SUM

- Apresentar qual o valor que a empresa gasta com salários atualmente

SELECT SUM(salario) FROM empregado



SUM

- Listar o valor que é gasto na folha de pagamento do setor financeiro

```
SELECT SUM(salario) FROM empregado  
WHERE setor='financeiro'
```




AVG

- Faz a média aritmética dos valores retornados em uma determinada coluna

SELECT AVG(coluna) FROM tabela



AVG

■ Apresentar a média de salário do setor financeiro

```
SELECT AVG(salario)  
FROM empregado  
WHERE setor='financeiro'
```