

JDBC

Alexandre Mendonça Fava
alexandre.fava@udesc.br

Universidade do Estado de Santa Catarina – UDESC
Programa de Pós-graduação em Computação Aplicada – PPGCA

Java DataBase Connectivity

Alexandre Mendonça Fava
alexandre.fava@udesc.br

Universidade do Estado de Santa Catarina – UDESC
Programa de Pós-graduação em Computação Aplicada – PPGCA

Conectividade Java com Banco de Dados

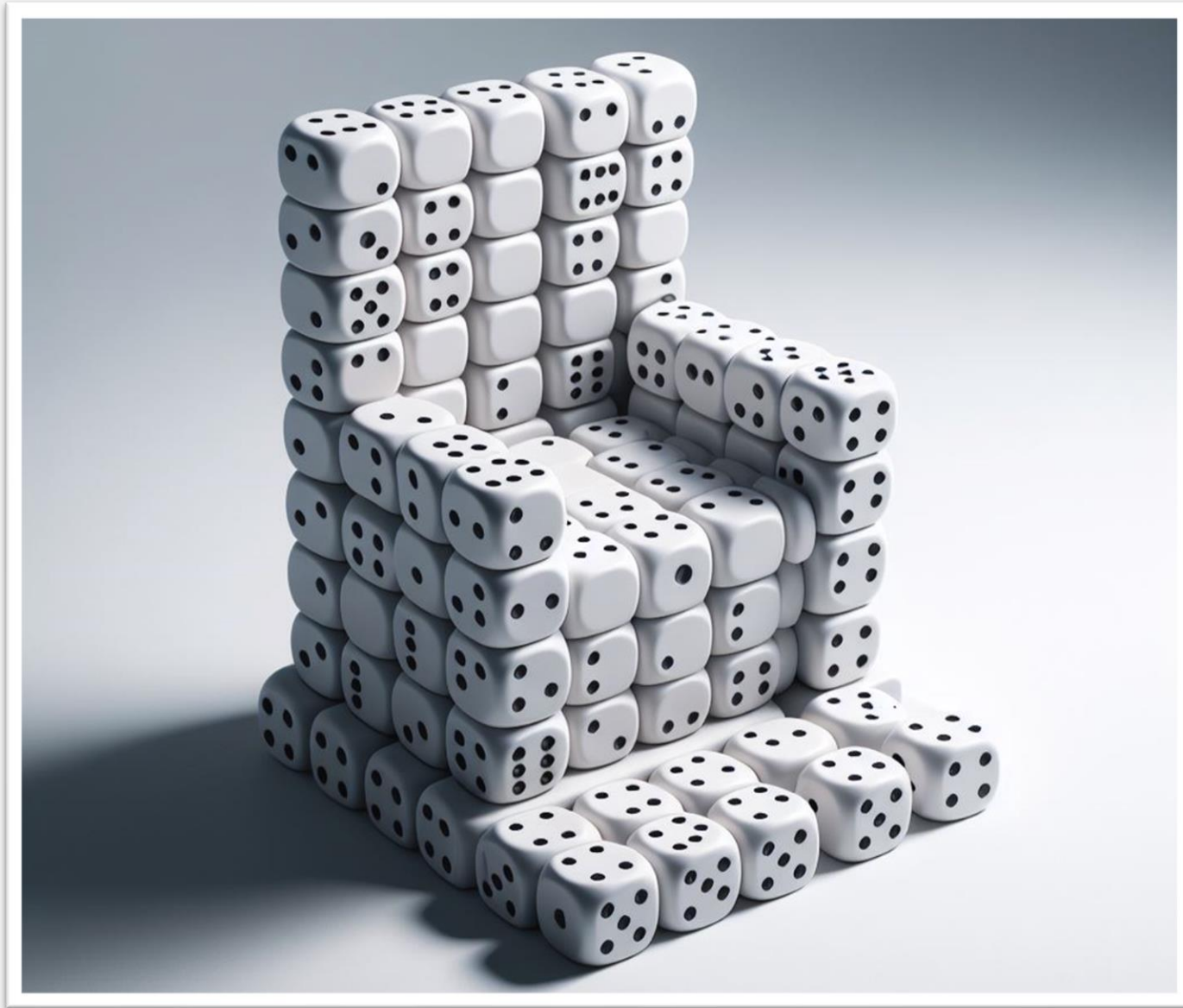
Alexandre Mendonça Fava
alexandre.fava@udesc.br

Universidade do Estado de Santa Catarina – UDESC
Programa de Pós-graduação em Computação Aplicada – PPGCA

Roteiro

- ☐ Definição
- ☐ Comandos
- ☐ Configuração
- ☐ Resumo
- ☐ Referências

Definição



Definição

Banco

banco \neq assento

Dados

Atributos ou Valores
(e.g.: 1, 2, 3, 4, 5, 6)
(e.g.: branco, preto)

dado \neq informação

Definição

Banco

Conjunto ou aglomerado
(e.g.: 1, 2, 3, 4, 5, 6)
(e.g.: branco, preto)

banco de dados = **bando** de dados

Dados

Atributos ou Valores
(e.g.: 1, 2, 3, 4, 5, 6)
(e.g.: branco, preto)

dado \neq informação

Definição

Banco

Conjunto ou aglomerado
(e.g.: números)
(e.g.: cores)

banco de dados = **bando** de dados

Dados

Atributos ou Valores
(e.g.: 1, 2, 3, 4, 5, 6)
(e.g.: branco, preto)

dado \neq informação

Definição

Banco de Dados

Coleção organizada de informações

DEITEL, Paul. Java: como programar.

24.1 Introdução

Um **banco de dados** é uma coleção organizada de dados. Há muitas estratégias diferentes para organizar dados para facilitar acesso e manipulação. Um **sistema de gerenciamento de bancos de dados** (database management system — DBMS) fornece mecanismos para armazenar, organizar, recuperar e modificar dados

Definição

Repositório de Impressões Digitais – FBI (1944)



Definição

Repositório Biométrico – FBI (2014)



Definição

Banco de Dados

Coleção organizada de informações

Local para o gerenciamento
e organização de dados



Definição

Banco de Dados

Coleção organizada de informações

Um local com dados desorganizados
não é um banco de dados



Definição

Banco de Dados

<https://www.flrs.co.uk/>

More data has been created in the past two years than in the entire history of humanity.

Fire and Security , Technology | 📅 January 27, 2021 | ✎ flr_wpadm

IT Support more crucial than ever, with more data being created in the past two years than in the entire previous history of humanity.

We live in an everchanging world of IT, technology advancing faster than the human brain is evolving. We live amongst devices that hold more information within them than we could ever store within our memory.

Definição

Banco de Dados

<https://www.flrs.co.uk/>

Mais dados foram criados nos últimos dois anos do que em toda a história da humanidade.

Fire and Security , Technology | 📅 January 27, 2021 | ✍ flr_wpadm

O suporte de TI é mais crucial do que nunca, com mais dados sendo criados nos últimos dois anos do que em toda a história anterior da humanidade.

Vivemos em um mundo em constante mudança de TI, a tecnologia avançando mais rápido do que o cérebro humano está evoluindo. Vivemos entre dispositivos que guardam mais informações dentro deles do que jamais poderíamos armazenar em

Definição

Banco de Dados



Definição

Banco de Dados

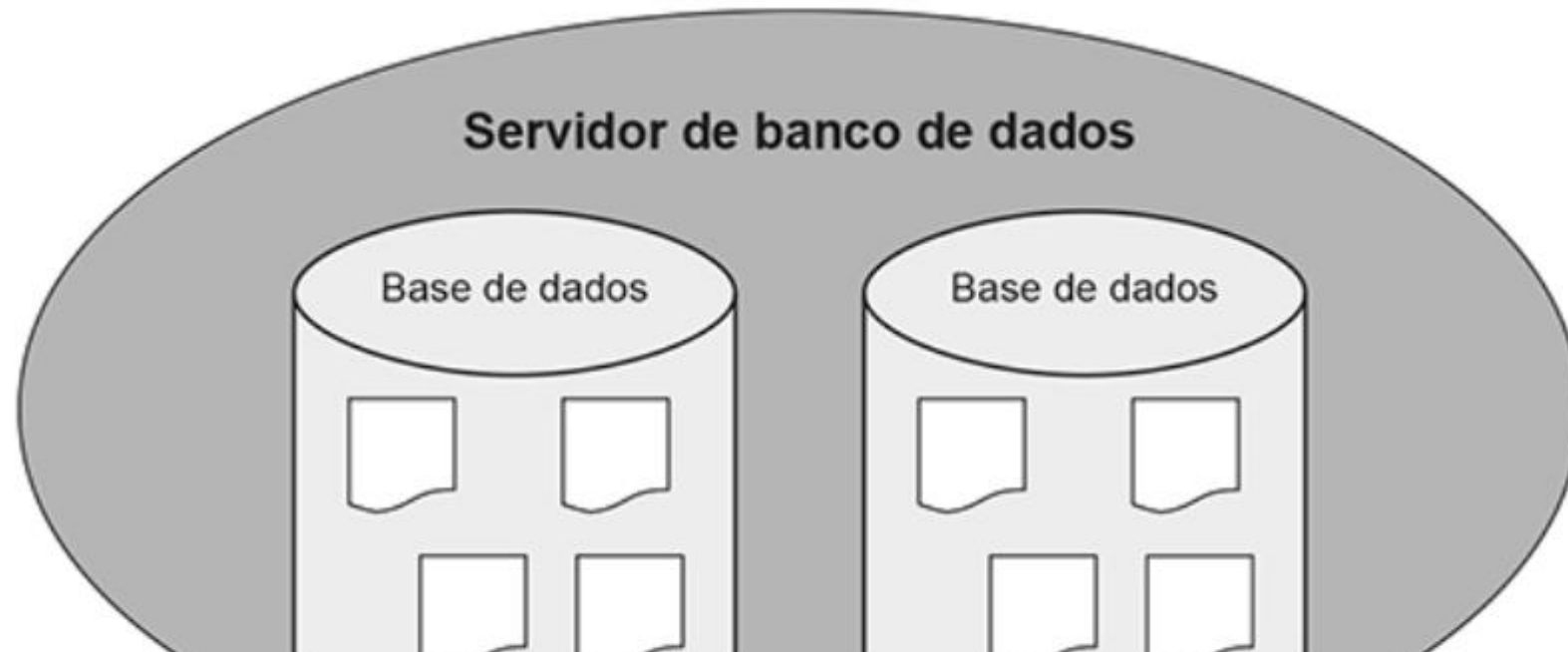
INFORMAÇÃO É PODER

facebook®

Google



Definição



Possibilita a independência entre seus dados e programas

Definição



Servidor de banco de dados

<https://www.flrs.co.uk/>

Banco de dados

🌐 100 línguas ▾

Artigo Discussão

Ler Editar Ver histórico Ferramentas ▾

Origem: Wikipédia, a enciclopédia livre.

Bancos de dados ^(português brasileiro) ou **bases de dados** ^(português europeu)^[1] são conjuntos de **arquivos** relacionados entre si, podendo conter registros sobre pessoas, lugares ou informações em geral. São coleções organizadas de **dados** que se relacionam ou não, de forma a armazenar informações.^{[2][3][4]}

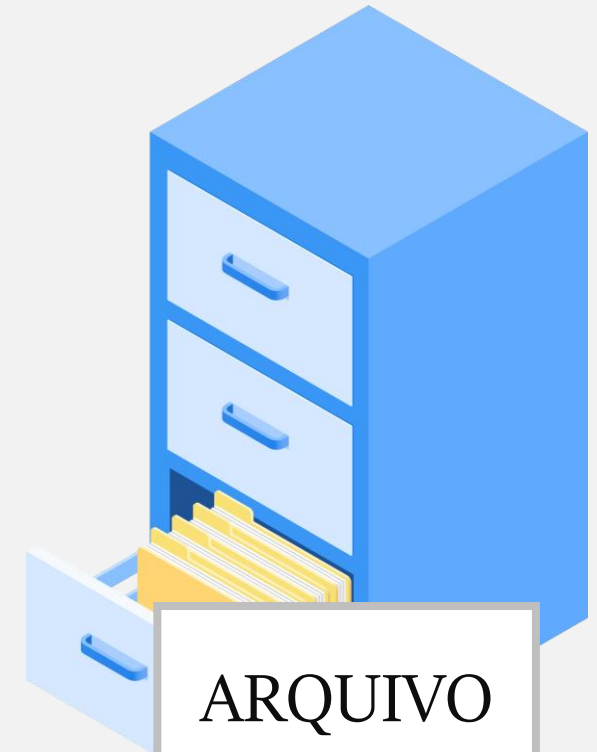
Definição

DADOS PESSOAIS									
Nome Completo					Comprador				
Carteira de Identidade	Orgão Carteira Profissional			Série	Outro documento				
CPF	Nacionalidade	Natural de		Data de Nascimento					
Filiação	Pai	Mãe							
Endereço Residencial				Nº	Apto.				
Bairro	Cidade	Estado		Telefone					
CEP	Casa Própria/Alugada/Com os pais/ Hotel/República/ Há quanto tempo								
Empresa que trabalha				Seção / Sala					
Endereço				Nº	Telefone				
Bairro	Cidade	Estado	Data admissão	Renda Mensal	Cargo				
Estado Civil	Nome do Esposo (a)		Profissão	Renda Mensal					
REFERÊNCIAS COMERCIAIS									
Referência					Referência				
Referência					Referência				
Pessoa Com Nome					Referência				
Endereço									

PAPEL



PASTA



ARQUIVO

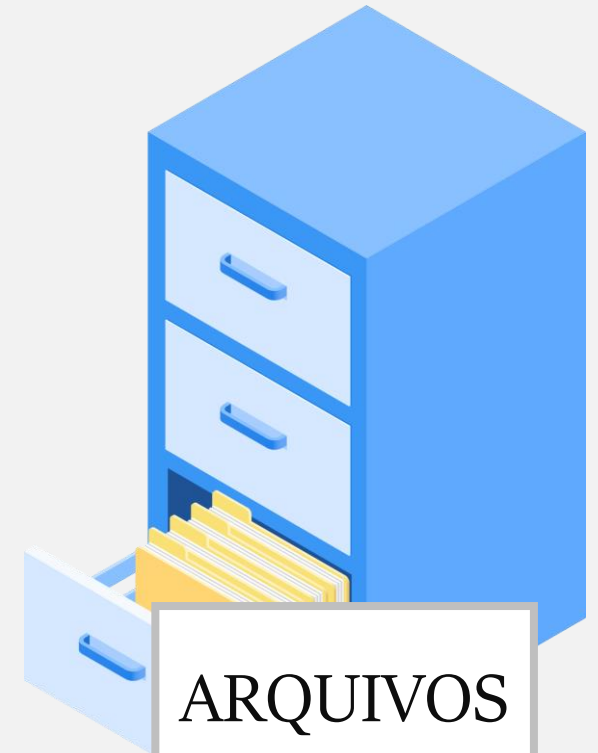
Definição

DADOS PESSOAIS									
Nome Completo					Comprador				
Carteira de Identidade	Orgão Carteira Profissional			Série	Outro documento				
CPF	Nacionalidade	Natural de		Data de Nascimento					
Filiação	Pai	Mãe							
Endereço Residencial				Nº	Apto.				
Bairro	Cidade	Estado		Telefone					
CEP	Casa Própria/Alugada/Com os pais/ Hotel/República/ Há quanto tempo								
Empresa que trabalha				Seção / Sala					
Endereço				Nº	Telefone				
Bairro	Cidade	Estado	Data admissão	Renda Mensal	Cargo				
Estado Civil	Nome do Esposo (a)		Profissão	Renda Mensal					
REFERÊNCIAS COMERCIAIS									
Referência					Referência				
Referência					Referência				
Pessoa Com Nome					Referência				
Endereço									

REGISTRO



TABELA



ARQUIVOS

Definição

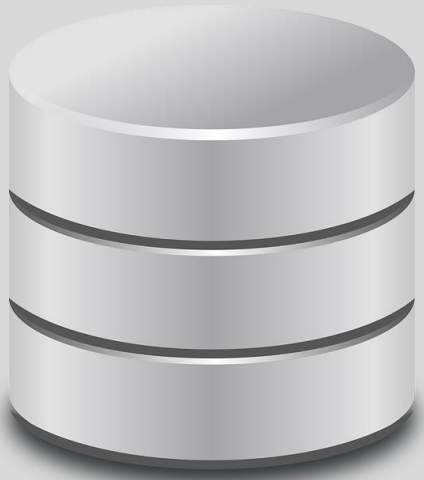


BANCO DE DADOS

Deve garantir:

- ☐ Padronização de acesso
- ☐ Segurança de acesso
- ☐ Integridade das informações
- ☐ Escalabilidade

Definição

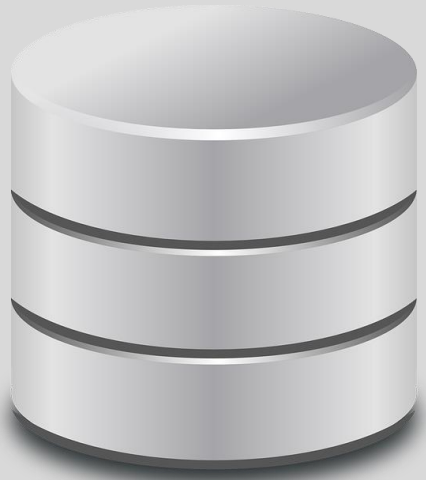


BANCO DE DADOS

Deve garantir:

- ☐ Mesmo programa
- ☐ Privilégio de acesso
- ☐ Histórico de modificações
- ☐ Bloquear ações conflitantes
- ☐ Mesmo tempo de acesso independente do tamanho

Definição



BANCO DE DADOS

Deve ter:



ARQUIVO

+



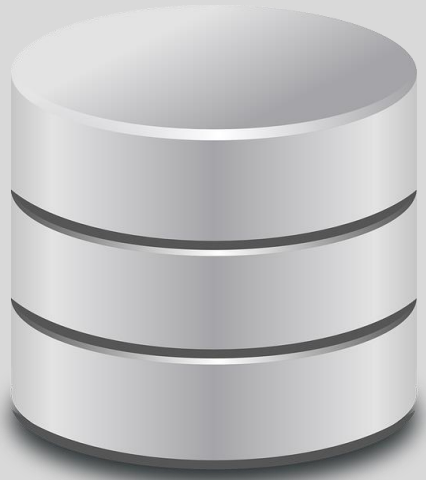
GERENCIADOR

+



LINGUAGEM

Definição



BANCO DE DADOS

Deve ter:



ARQUIVO

+



GERENCIADOR

+



LINGUAGEM

Definição

DBMS

NÃO É UMA IDE
É UM PROGRAMA

GRATUITO



PAGO



ORACLE®

SQL

NÃO É UMA LINGUAGEM DE
PROGRAMAÇÃO
É UMA LINGUAGEM DE
CONSULTA OU EXPLORAÇÃO

É uma linguagem padronizada...
Com vários dialetos

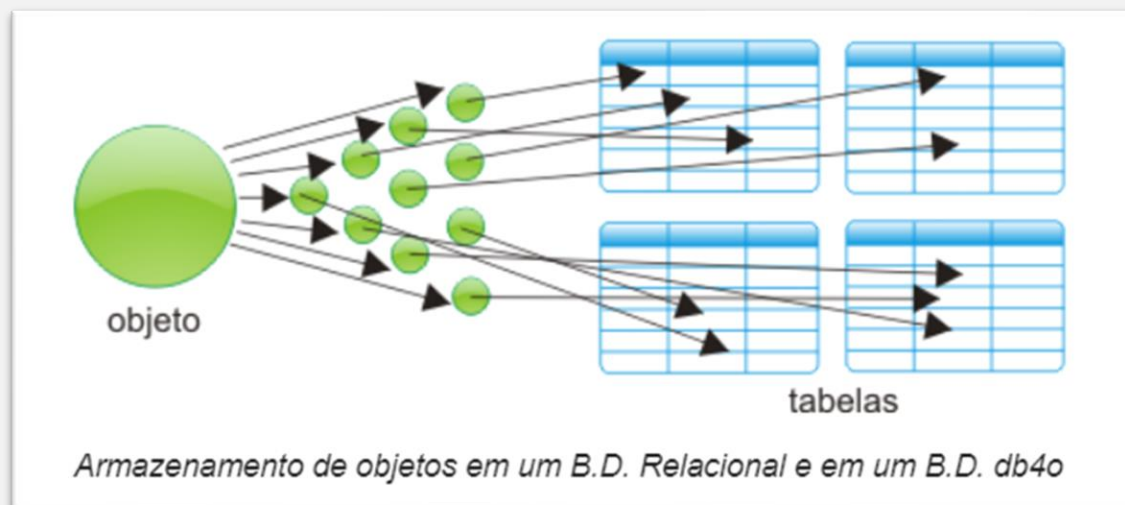
Dialects

SQL-86 · SQL-89 · SQL-92 · SQL:1999 ·
SQL:2003 · SQL:2006 · SQL:2008 · SQL:2011
· SQL:2016 · SQL:2023

Um Sistema de gerenciamento de banco de dados (DBMS) fornece mecanismo para armazenar, organizar, recuperar e modificar dados.



Definição



DEITEL, Paul. Java: como programar.

Os sistemas de gerenciamento de bancos de dados mais populares fornecem drivers JDBC. Neste capítulo, introduzimos o JDBC e o utilizamos para manipular bancos de dados Java DB. As técnicas demonstradas aqui também podem ser utilizadas para manipular outros bancos de dados que têm drivers JDBC.

Definição

DEITEL, Paul. Java: como programar.

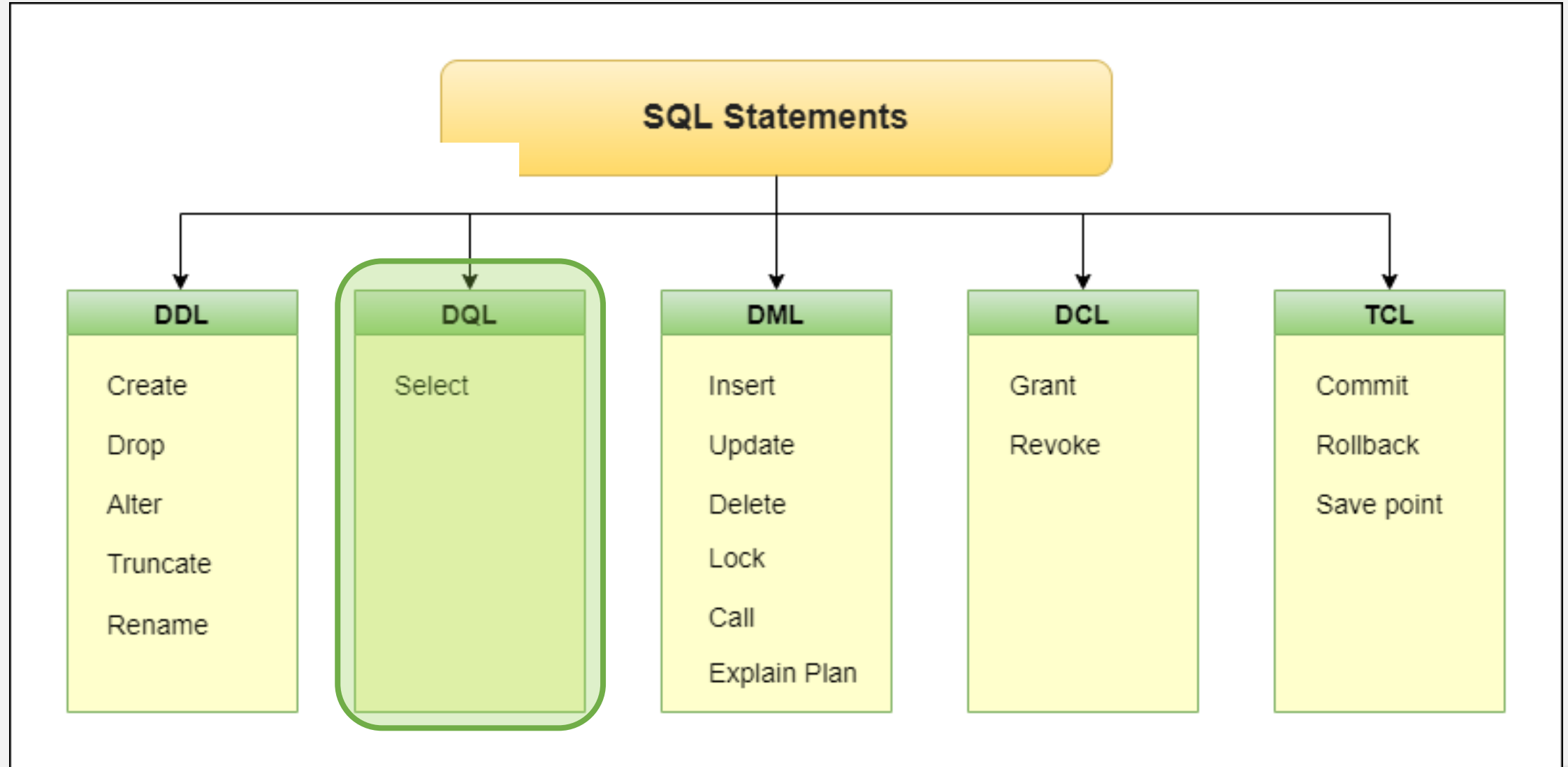
RDBMS	Formato de URL de banco de dados
MySQL	<code>jdbc:mysql:// nomeDoHost:númeroDePorta/nomeDoBancoDeDados</code>
ORACLE	<code>jdbc:oracle:thin:@ nomeDoHost:númeroDePorta:nomeDoBancoDeDados</code>
DB2	<code>jdbc:db2: nomeDoHost:númeroDePorta/nomeDoBancoDeDados</code>
PostgreSQL	<code>jdbc:postgresql:// nomeDoHost:númeroDePorta/nomeDoBancoDeDados</code>
Java DB/Apache Derby	<code>jdbc:derby:nomeDoBancoDados</code> (incorporado)
	<code>jdbc:derby://nomeDoHost:númeroDaPorta/nomeDoBancoDeDados</code> (rede)
Microsoft SQL Server	<code>jdbc:sqlserver://nomeDoHost: númeroDaPorta;databaseName=nomeDoBancoDeDados</code>
Sybase	<code>jdbc:sybase:Tds: nomeDoHost:númeroDePorta/nomeDoBancoDeDados</code>

Figura 24.24 | Formatos URL populares do banco de dados JDBC.

SQL

Divisões	Definição	Comandos
DML (Data Manipulation Language)	Comandos que inserem, atualizam ou excluem dados nas tabelas	INSERTDELETEUPDATE
DDL (Data Definition Language)	Comandos que criam, alteram ou excluem objetos do banco de dados	CREATEALTERDROP
DCL (Data Control Language)	Comandos que gerenciam as permissões e privilégios dos usuários do banco de dados	GRANTREVOKEDENY
DTL (Data Transaction Language)	Comandos que controlam as transações realizadas no banco de dados	COMMITROLLBACK
DQL (Data Query Language)	Comandos que permitem consultar os dados armazenados nas tabelas do banco de dados	SELECT

SQL



Comando (Bando de Dados)

Loja-BD

Consumidores

Consumidor_ID	Nome	Sobrenome	Idade	Pais
1	John	Doe	31	USA
2	Robert	Luna	22	USA
3	David	Robinson	22	UK
4	John	Reinhardt	25	UK
5	Betty	Doe	28	UAE

Pedidos

Pedido_ID	Item	Quantidade	Consumidor_ID
1	Keyboard	400	4
2	Mouse	300	4
3	Monitor	12000	3
4	Keyboard	400	1
5	Mousepad	250	2

Comando: Select

SINTAXE

SELECT coluna **FROM** tabela [**WHERE** condição];

OPERAÇÃO

Realiza uma consulta em uma ou mais tabelas

Comando: Select

COMANDO

```
SELECT nome FROM Consumidores;
```

RESULTADO

Nome
John
Robert
David
John
Betty

Comando: Select

COMANDO

```
SELECT nome, sobrenome  
FROM Consumidores;
```

RESULTADO

Nome	Sobrenome
John	Doe
Robert	Luna
David	Robinson
John	Reinhardt
Betty	Doe

Comando: Select

COMANDO

```
SELECT nome AS "Coluna1",  
sobrenome AS "Coluna2"  
FROM Consumidores;
```

RESULTADO

Coluna1	Coluna2
John	Doe
Robert	Luna
David	Robinson
John	Reinhardt
Betty	Doe

Comando: Select

COMANDO

```
SELECT * FROM Consumidores;
```

RESULTADO

Consumidor_ID	Nome	Sobrenome	Idade	Pais
1	John	Doe	31	USA
2	Robert	Luna	22	USA
3	David	Robinson	22	UK
4	John	Reinhardt	25	UK
5	Betty	Doe	28	UAE

Comando: Count

COMANDO

```
SELECT COUNT(*) FROM Consumidores;
```

RESULTADO

COUNT(*)

5

Comando: Select .. Where

COMANDO

```
SELECT * FROM Consumidores  
WHERE idade=22;
```

RESULTADO

Consumidor_ID	Nome	Sobrenome	Idade	Pais
2	Robert	Luna	22	USA
3	David	Robinson	22	UK

Comando: Select .. Where

COMANDO

```
SELECT * FROM Consumidores  
WHERE Idade IN (22)
```

RESULTADO

Consumidor_ID	Nome	Sobrenome	Idade	Pais
2	Robert	Luna	22	USA
3	David	Robinson	22	UK

Comando: Select .. Where .. And

COMANDO

```
SELECT * FROM Consumidores  
WHERE idade=22 AND pais="USA";
```

RESULTADO

Consumidor_ID	Nome	Sobrenome	Idade	Pais
2	Robert	Luna	22	USA

Comando: Select .. Where .. Or

COMANDO

```
SELECT * FROM Consumidores  
WHERE idade=42 OR pais="USA";
```

RESULTADO

Consumidor_ID	Nome	Sobrenome	Idade	Pais
1	John	Doe	31	USA
2	Robert	Luna	22	USA

Comando: Order by

COMANDO

```
SELECT * FROM Consumidores  
ORDER BY nome;
```

RESULTADO

Consumidor_ID	Nome	Sobrenome	Idade	Pais
5	Betty	Doe	28	UAE
3	David	Robinson	22	UK
1	John	Doe	31	USA
4	John	Reinhardt	25	UK
2	Robert	Luna	22	USA

Comando: Avg

COMANDO

```
SELECT AVG(idade) FROM Consumidores;
```

RESULTADO

AVG(idade)

25.6

Comando: Min

COMANDO

```
SELECT MIN(idade) FROM Consumidores;
```

RESULTADO

MIN(idade)
31

Comando: Max

COMANDO

```
SELECT MAX(idade) FROM Consumidores;
```

RESULTADO

MAX(idade)

31

Comando: Sum

COMANDO

```
SELECT SUM(idade) FROM Consumidores;
```

RESULTADO

SUM(idade)
128

Comando: Sum .. Group by

COMANDO

```
SELECT SUM(idade), pais  
FROM Consumidores  
GROUP BY (pais);
```

RESULTADO

SUM(idade)	pais
28	UAE
47	UK
53	USA

Comando: Sum .. Group by .. Having

COMANDO

```
SELECT SUM(idade), pais  
FROM Consumidores  
GROUP BY pais  
HAVING AVG(idade) > 25;
```

RESULTADO

SUM(idade)	pais
28	UAE
53	USA

Comando: Sum .. Order by

COMANDO

```
SELECT SUM(idade), pais  
FROM Consumidores  
ORDER BY (pais);
```

RESULTADO

SUM(idade)	pais
128	USA

Comando: Between

COMANDO

```
SELECT Item, Quantidade  
FROM Pedidos  
WHERE Quantidade  
BETWEEN 300 AND 1000;
```

RESULTADO

Item	Quantidade
Keyboard	400
Mouse	300
Keyboard	400

Comando: Distinct

COMANDO

```
SELECT DISTINCT(Item), Quantidade  
FROM Pedidos  
WHERE Quantidade  
BETWEEN 300 AND 1000;
```

RESULTADO

Item	Quantidade
Keyboard	400
Mouse	300

Comandos de Diferentes Tabelas

COMANDO

```
SELECT Nome, Item
FROM Consumidores, Pedidos
WHERE Consumidores.Consumidor_ID=Pedidos.Consumidor_ID;
```

RESULTADO

nome	Item
John	Keyboard
John	Mouse
David	Monitor
John	Keyboard
Robert	Mousepad

Comandos

COMANDO

```
SELECT Nome, Item  
FROM Consumidores, Pedidos;
```

John	Monitor
John	Keyboard
John	Mousepad
Robert	Keyboard
Robert	Mouse
Robert	Monitor
Robert	Keyboard
Robert	Mousepad
David	Keyboard
David	Mouse
David	Monitor
David	Keyboard
David	Mousepad
John	Keyboard
John	Mouse

Comando: Like

COMANDO

```
SELECT Sobrenome, Idade  
FROM Consumidores  
WHERE Sobrenome LIKE "R%"
```

RESULTADO

Sobrenome	Idade
Robinson	22
Reinhardt	25

Comando: Like

COMANDO

```
SELECT Sobrenome, Idade  
FROM Consumidores  
WHERE Sobrenome LIKE "%e"
```

RESULTADO

Sobrenome	Idade
Doe	31
Doe	28

Comando: Like

COMANDO

```
SELECT Sobrenome, Idade  
FROM Consumidores  
WHERE Sobrenome LIKE "_o%"
```

RESULTADO

Sobrenome	Idade
Doe	31
Robinson	22
Doe	28

Não visto...

- Modelo Relacional
- Modelo Entidade Relacionamento
- Álgebra Relacional
- Cálculo Relacional
- Chave-primária e chave-estrangeira

Exceções

As exceções ocorrem quando algo imprevisto acontece, elas podem ser provenientes de erros de lógica ou acesso a recursos que talvez não estejam disponíveis.

Principais causas:

- Tentar abrir um arquivo que não existe.
- **Tentar fazer consulta a um banco de dados que não está disponível.**
- Tentar escrever algo em um arquivo sobre o qual não se tem permissão de escrita.
- **Tentar conectar em servidor inexistente.**
- Tentar manipular um objeto que está com o valor nulo.
- **Dividir um número por zero.**
- Tentar manipular um tipo de dado como se fosse outro.
- Tentar utilizar um método ou classe não existentes.

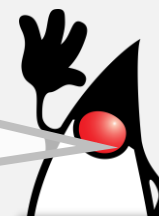
Exceções

Código (Main.java)

```
1 public class Main{
2     public static void main(String[] args){
3         int a = 10, b = 0;
4         try{
5             int c = a/b;
6             System.out.println("Resultado: " + c);
7         }catch(ArithmeticException excecao){
8             System.out.println("Erro: " + excecao);
9         }finally{
10            System.out.println("Terminei");
11        }
12    }
13 }
14 }
```

Ei, isso está me parecendo uma: IF ELSE.

Embora ambos façam o controle de fluxo, o TRY CATCH é voltado para resolver exceções, tratando ou propagando (throw).



Exceções

Código (Main.java)

```
1 public class Main{
2     public static void main(String[] args){
3         int a = 10, b = 0;
4         try{
5             int c = a/b;
6             System.out.println("Resultado: " + c);
7         }catch(ArithmeticException excecao){
8             System.out.println("Erro: " + excecao);
9         }finally{
10             System.out.println("Terminei");
11         }
12     }
13 }
```

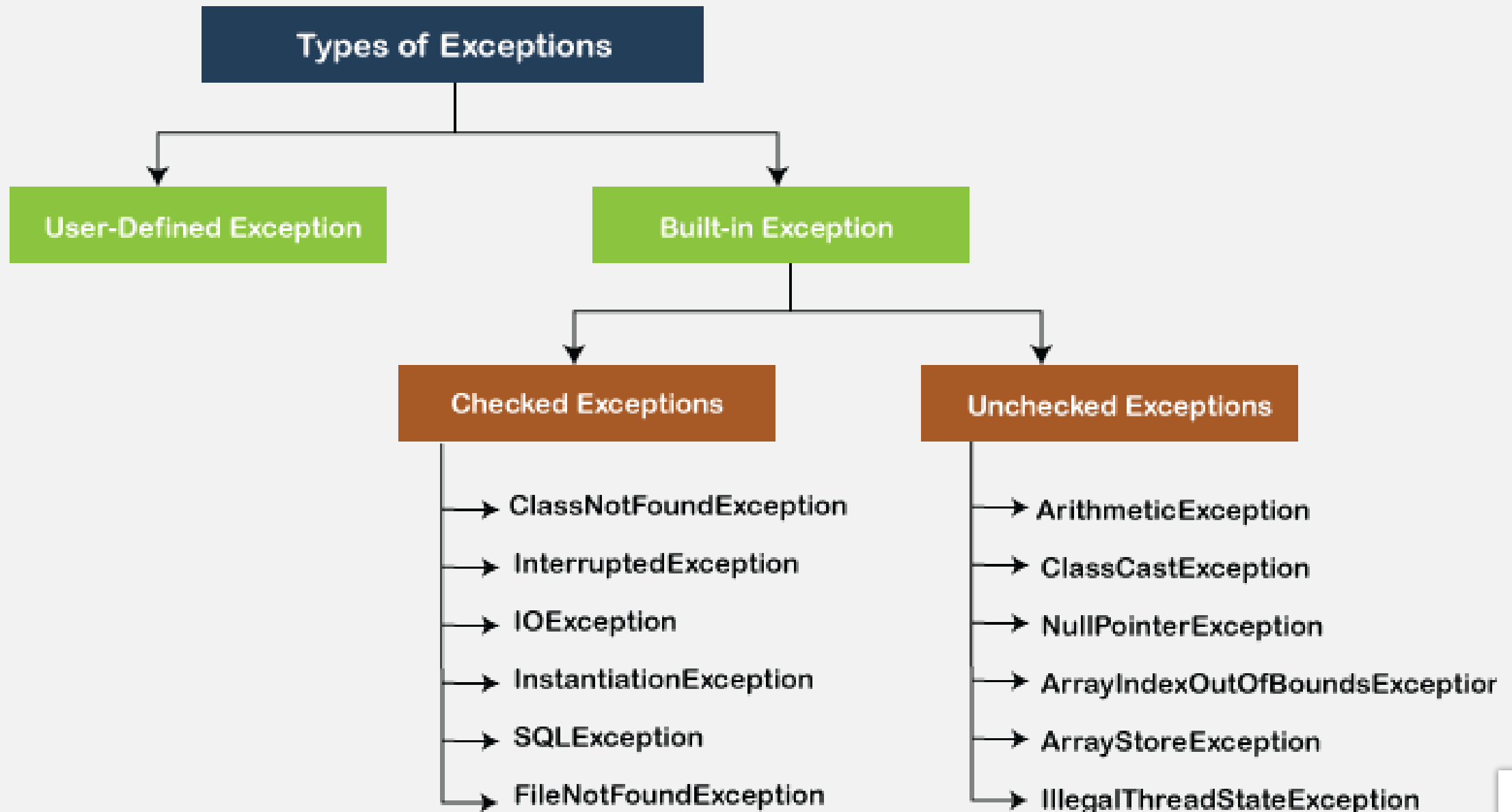
TENTA
EXECUTAR

SEMPRE
EXECUTA

CAPTURA
EXCEÇÃO

É possível ter vários catch

Exceções



Preparando a casa

<https://docs.oracle.com/>

Lesson: JDBC Introduction

The JDBC API is a Java API that can access any kind of tabular data, especially data stored in a relational database.

JDBC helps you to write Java applications that manage these three programming activities:

1. Connect to a data source, like a database
2. Send queries and update statements to the database
3. Retrieve and process the results received from the database in answer to your query

The following simple code fragment gives a simple example of these three steps:

```
public void connectToAndQueryDatabase(String username, String password) {
```

Preparando a casa

Código (Main.java)

```
1 public class Main{
2     public static void main(String[] args){
3         try{
4             Class.forName("com.mysql.jdbc.Driver");
5             Connection con = DriverManager.getConnection("url",
6 "usuario", "senha");
7             ResultSet rs =
8 con.createStatement().executeQuery("SELECT * FROM Tabela");
9         }catch(ArithmeticException excecao){
10             System.out.println("Erro: " + excecao);
11         }finally{
12             System.out.println("Terminei");
13             con.close()
14         }
```

Carrega o drive do banco de dados

Autenticação

JDBC

Alexandre Mendonça Fava
alexandre.fava@udesc.br

Universidade do Estado de Santa Catarina – UDESC
Programa de Pós-graduação em Computação Aplicada – PPGCA