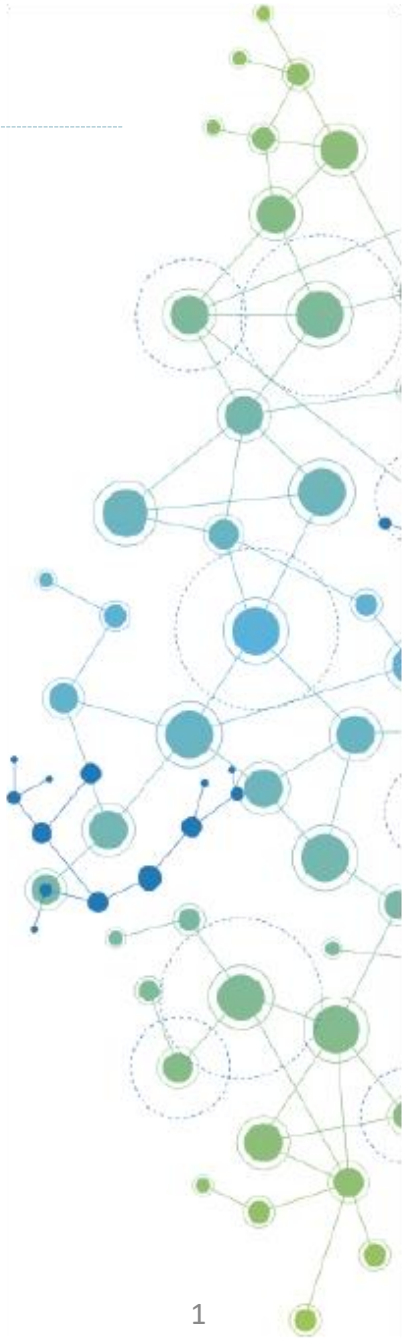


Introdução ao SQL:

Funções importantes para a Análise de Dados

Extração de Dados

Funções importantes para a Análise de Dados

Preditiva.ai

Quando possuímos **um volume muito grande de informações** e temos a nossa disposição apenas um computador comum, com processamento e memória bastante limitados, pode ser interessante realizar parte das análises no **servidor de banco de dados**.

A seguir veremos como podemos realizar algumas das etapas de uma **análise exploratória** utilizando comandos SQL:

1. Tabelas de **frequência absoluta** e **relativa**

2. **Medidas Resumo**

Extração de Dados

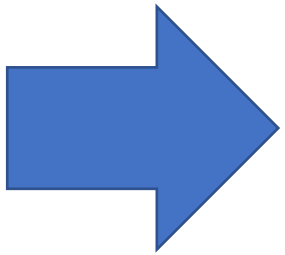
Funções importantes para a Análise de Dados: COUNT

Preditiva.ai

Uma forma de resumir os dados resultantes de uma consulta é utilizar as funções de agregação. Elas possibilitam resumir dados de múltiplas observações.

Suponha que você deseje saber quantos passageiros haviam no Titanic. Para isso utilizaremos a função de agregação **COUNT**:

```
SELECT
    COUNT(PassengerId)
FROM
    titanic
```



	No Column Name
	INT
1	891

JOAO ESTEVAN LEONCIO DA SILVA BARBOSA - jestevan12@gmail.com - CPF: 134.982.877-70

Extração de Dados

Funções importantes para a Análise de Dados: COUNT

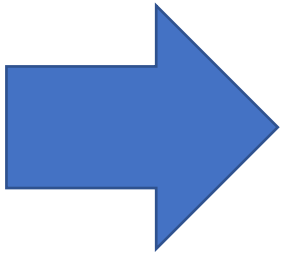
Preditiva.ai

A função **COUNT** pode ser utilizada de duas diferentes formas.

1. Contar o número de registros na tabela: COUNT(1)
2. Contar o número de registros com valores não nulos em determinado campo: COUNT(campo)

Esta 2ª forma é bastante útil para identificar campos que possuem **NULL**, ou *missing values*.

```
SELECT
    COUNT(1),
    COUNT(PassengerId),
    COUNT(Age)
FROM
    titanic
```



	No Column Name	No Column Name	No Column Name
	INT	INT	INT
1	891	891	714

Com a consulta acima extraímos as seguintes informações:

1. A tabela possui 891 registros
2. O campo PassengerId possui 891 valores válidos (não nulos)
3. O campo Age possui 714 valores válidos, ou seja 177 dos 891 são nulos

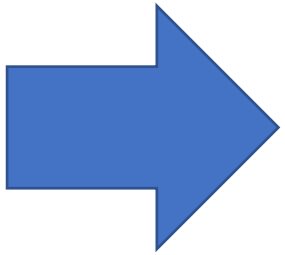
Extração de Dados

Funções importantes para a Análise de Dados: Funções Agregação

Preditiva.ai

Uma outra informação que pode ser importante nesse conjunto de dados é o valor pago nas passagens. Podemos calcular algumas medidas resumo deste campo utilizando as funções de agregação **MIN**, **MAX**, **SUM**, **AVG** e **STDEV**:

```
SELECT
    COUNT(passengerId),
    MIN(Fare),
    MAX(Fare),
    SUM(Fare),
    AVG(Fare),
    STDEV(Fare)
FROM
    titanic
```



	No Column Name	No Column Name	No Column Name	No Column Name	No Column Name	No Column Name
	INT	FLOAT	FLOAT	FLOAT	FLOAT	FLOAT
1	891	0	5.123.292	113.745.644	127.660,65544332	411.122,99261142

Apesar do resultado trazer as informações que solicitamos, os nomes das colunas ficaram todos iguais. Vamos aprender como melhorar isso usando **ALIAS**.

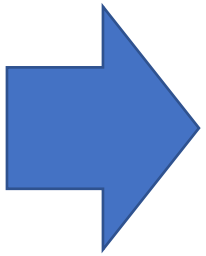
Extração de Dados

Funções importantes para a Análise de Dados: Funções Agregação

Preditiva.ai

O **ALIAS** serve para renomearmos os campos, ou nesse caso atribuir um nome após a aplicação da função de agregação.
Com isso, a informação fica mais organizada e fácil de se analisar.

```
SELECT
    COUNT(passengerId) as n,
    MIN(Fare) as tarifa_min,
    MAX(Fare) as tarifa_max,
    SUM(Fare) as tarifa_total,
    AVG(Fare) as tarifa_media,
    STDEV(Fare) as tarifa_dp
FROM
    titanic
```



	n	tarifa_min	tarifa_max	tarifa_total	tarifa_media	tarifa_dp
	INT	FLOAT	FLOAT	FLOAT	FLOAT	FLOAT
1	891	0	5.123.292	113.745.644	127.660,65544332	411.122,99261142

Podemos perceber que o desvio padrão das tarifas, de aproximadamente 411 mil, é bastante grande quando comparado com a média. Isso pode ser porque estamos calculando essas medidas resumo de todos os passageiros de diferentes classes. Vamos então ver como calcular essas medidas resumo por classe.

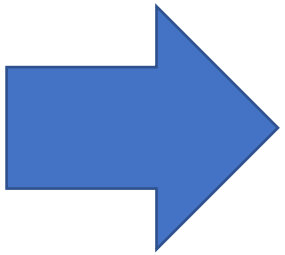
Extração de Dados

Funções importantes para a Análise de Dados: GROUP BY

Preditiva.ai

O **GROUP BY** serve para realizarmos as agregações agrupadas pelos valores de um ou mais campos.
Vamos calcular as mesmas medidas resumo, agora agrupadas e ordenadas por classe:

```
SELECT
    Pclass,
    COUNT(passengerId) as n,
    MIN(Fare) as tarifa_min,
    MAX(Fare) as tarifa_max,
    SUM(Fare) as tarifa_total,
    AVG(Fare) as tarifa_media,
    STDEV(Fare) as tarifa_dp
FROM
    titanic
GROUP BY
    Pclass
ORDER BY
    Pclass
```



	Pclass	n	tarifa_min	tarifa_max	tarifa_total	tarifa_media	tarifa_dp
	TINYINT	INT	FLOAT	FLOAT	FLOAT	FLOAT	FLOAT
1	1	216	0	5.123.292	82.946.250	384.010,41666667	766.130,98810115
2	2	184	0	415.792	3.524.792	19.156,47826087	76.227,79571568
3	3	491	0	564.958	27.274.602	55.549,08757637	93.204,37904102

O desvio padrão total que era aproximadamente 411 mil aumentou para 766 mil na 1ª classe e diminuiu para 76 mil e 93 mil na 2ª e 3ª classes, respectivamente.

Preditiva.ai