

```

1  /*
2  Banco de Dados
3  Prof.ª Mariana Meirelles de Mello
4  Semestre: 2025.2
5  Data: 18/11/2025
6
7  Aula 05: Funções de Agregação, GROUP BY e HAVING
8
9  Aprender a fazer consultas que trazem informações
10 resumidas: totais, médias, contagens etc.
11 */
12
13 /*
14 FUNÇÕES DE AGREGAÇÃO:
15
16 COUNT() - Conta registros
17 SUM() - Soma valores numéricos
18 AVG() - Calcula a média
19 MAX() - Maior valor
20 MIN() - Menor valor
21 */
22
23 -- Quantos moradores existem no total?
24 SELECT COUNT(*) AS 'Qtd Morador'
25 FROM morador;
26
27 -- Quantos bairros existem cadastrados?
28 SELECT count(*) AS 'Qtd Bairro'
29 FROM bairro;
30
31 -- Qual é a maior população entre os bairros?
32 SELECT MAX(populacao) AS 'Maior Populacao'
33 FROM bairro;
34
35 -- Qual é a média de população dos bairros?
36 SELECT AVG(populacao) AS 'Média População'
37 FROM bairro;
38
39
40 SELECT round(AVG(populacao),1) AS 'Média População'
41 FROM bairro;
42 /* A função round(valor, casas_decimais) arredonda um valor passado como parâmetro
43 de acordo com a quantidade de casas decimais desejada. Se não colocar casas decimais,
44 ele irá resultar em um número inteiro. */
45
46 -- Quantos eventos existem cadastrados?
47 SELECT COUNT(*) AS 'Qtd Eventos'
48 FROM evento;
49
50 /* GROUP BY (Agrupamento)
51 Mostra os totais por categoria, agrupando dados.
52 O GROUP BY é usado em SQL para agrupar registros que possuem valores iguais
53 em uma ou mais colunas.
54 Ele permite transformar vários registros individuais em grupos, sobre os quais
55 podemos aplicar funções de agregação.
56 */
57
58 -- Quantidade de eventos por ano
59 SELECT year(data) AS 'Ano', count(*) AS 'Qtd'
60 FROM evento
61 GROUP BY year(data);
62
63 SELECT *
64 FROM evento;
65
66 -- Como só tínhamos eventos no ano de 2025, fizemos alguns updates para algumas datas
67 em 2024 e 2026.
68 UPDATE evento
69 SET data = '2026-03-20'
70 WHERE idevento IN (1,5);
71
72 -- Após os Updates, um novo select para verificar como ficou nosso resultado agora
73 SELECT year(data) AS 'Ano', count(*) AS 'Qtd'

```

```

73  FROM evento
74  GROUP BY year(data);
75
76  -- Quantidade de eventos por mês
77  SELECT month(data) AS 'Mês', count(*) as 'Qtd'
78  FROM evento
79  GROUP BY month(data);
80
81  /*Mesma consulta usando o ORDER BY, ele tem que estar após o GROUP BY*/
82  SELECT month(data) AS 'Mês', count(*) as 'Qtd'
83  FROM evento
84  GROUP BY month(data)
85  ORDER BY month(data);
86
87  -- Contar moradores por código do bairro
88  SELECT count(*) AS 'Qtd', fk_idbairro AS 'Codigo Bairro'
89  FROM morador
90  GROUP BY fk_idbairro;
91
92  /*
93  HAVING
94
95  O HAVING é usado para filtrar grupos depois que eles já
96  foram formados pelo GROUP BY.
97
98      - Ele não filtra linhas individuais.
99      - Ele filtra grupos inteiros.
100
101 ATENÇÃO! Sempre usado junto de GROUP BY
102
103 WHERE X HAVING
104
105     WHERE - Filtra linhas.
106     HAVING - Filtra grupos.
107 */
108
109  -- Mostrar meses que têm mais de 1 evento
110  SELECT month(data) AS 'Mês', COUNT(*) AS 'Qtd'
111  FROM evento
112  GROUP BY month(data)
113  HAVING COUNT(*) > 1;
114
115  -- Mostrar bairros (apenas o código) com mais de 1 morador
116  SELECT fk_idbairro AS 'Num Bairro', COUNT(*) AS 'Qtd'
117  FROM morador
118  GROUP BY fk_idbairro
119  HAVING COUNT(*) > 1;
120
121  -- Mostrar anos que tiveram mais de 2 eventos
122  SELECT year(data) AS 'Ano Evento', COUNT(*) AS 'Qtd'
123  FROM evento
124  GROUP BY year(data)
125  HAVING COUNT(*) > 2
126  ORDER BY year(data);

```