

# CONSULTANDO DADOS

```
SELECT *  
  FROM alunosBetha;
```

```
SELECT i_alunos, nome, cidade  
  FROM alunosBetha;
```

# ALIAS

```
SELECT i_alunos as cdAluno,  
       nome as nmAluno,  
       cidade as nmCidade  
FROM alunosBetha;
```

# FILTRANDO DADOS

```
SELECT *  
  FROM alunosBetha  
 WHERE i_alunos = 1;
```



# DISTINCT

```
SELECT DISTINCT cidade  
FROM alunosBetha;
```

# ORDENAÇÃO

```
SELECT *  
  FROM alunosBetha  
 ORDER BY i_alunos asc;
```

# OPERADORES LÓGICOS

AND

OR

NOT



# BETWEEN

```
SELECT *  
  FROM alunosBetha  
 WHERE i_alunos between 3 and 5;
```

# COMPUTADOS

```
SELECT i_alunos,  
       i_alunos + 1 as prox,  
       3 + 4 as sete  
FROM alunosBetha;
```



# FUNÇÕES STRING

||

length()

lower

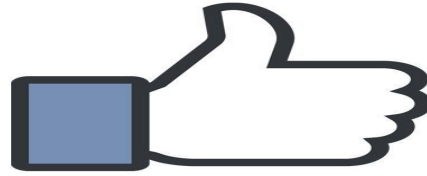
upper

trim

substr

replace

LIKE



```
SELECT *  
  FROM alunosBetha  
 WHERE nome like '_iago%'
```

# ATUALIZANDO DADOS (COM WHERE)

```
UPDATE alunosBetha  
    SET nome = 'Outro Aluno'  
WHERE i_alunos = 1;
```



# TRANSAÇÕES

```
BEGIN;  
UPDATE alunosBetha  
    SET nome = 'TESTE 1'  
WHERE i_alunos = 1;  
UPDATE alunosBetha  
    SET nome = 'TESTE 2'  
WHERE i_alunos = 2;  
COMMIT;
```

# CASE

```
SELECT  
    CASE WHEN media < 5 THEN 'Reprovado'  
          WHEN media < 7 THEN 'Recuperação'  
          ELSE 'Aprovado' END  
FROM alunos;
```

# COALESCE

```
SELECT i_alunos,  
       nome,  
       COALESCE(cidade,'Cidade não informada')  
FROM alunosBetha;
```



# UNION

```
SELECT 'Risco A' as tp, descricao as nm  
FROM atividades WHERE aliquota > 5
```

UNION

```
SELECT 'Risco B' as tp, descricao as nm  
FROM atividades WHERE aliquota < 2
```

# JUNÇÃO DE TABELAS

```
SELECT *  
  FROM atividades, empresas  
 WHERE atividades.i_atividades = empresas.i_atividades
```

```
SELECT *  
  FROM atividades INNER JOIN empresas  
    ON (atividades.i_atividades = empresas.i_atividades)
```

# AGRUPAMENTO

```
SELECT cidade  
  FROM alunosBetha  
GROUP BY cidade
```



# FUNÇÕES AGRUPAMENTO

COUNT

SUM

AVG

MAX

MIN

# HAVING

```
SELECT tipo_pessoa, COUNT(*)  
  FROM pessoas  
 GROUP BY tipo_pessoa  
HAVING COUNT(*) > 1
```

# SUBCONSULTA

```
SELECT i_atividades AS cdAtiv,  
      (SELECT count(*)  
        FROM empresas  
        WHERE i_atividades = atividades.i_atividades )  
FROM atividades;
```



# EXISTS

```
SELECT *  
  FROM pessoas  
 WHERE EXISTS(SELECT 1  
              FROM enderecos_pessoas  
              WHERE enderecos_pessoas.i_pessoas  
                    = pessoas.i_pessoas)
```

# IN / NOT IN

```
SELECT *  
  FROM pessoas  
WHERE i_pessoas in (SELECT i_pessoas  
                    FROM enderecos_pessoas)
```