

Lista de exercícios – W3Schools

Lucas Mendonça da Silva Junior

SQL Select

Ex01

SQL Select

✓ Exercise 1

✓ Exercise 2

Exercise 3

[Go to SQL Select Tutorial](#)

SQL Where

SQL Order By

SQL Insert

SQL Null

SQL Update

SQL Delete

Exercise:

Insert the missing statement to get all the columns from the `Customers` table.

Correct!

Next >

Next Exercise >

SELECT

*

FROM

Customers;

Ex02

SQL Select

✓ Exercise 1

✓ Exercise 2

Exercise 3

[Go to SQL Select Tutorial](#)

SQL Where

SQL Order By

SQL Insert

SQL Null

SQL Update

SQL Delete

Exercise:

Write a statement that will select the `City` column from the `Customers` table.

Correct!

Next >

Next Exercise >

SELECT

City

FROM

Customers;

Ex03

SQL Select ✓

✓ Exercise 1

✓ Exercise 2

✓ Exercise 3

[Go to SQL Select Tutorial](#)

SQL Where

SQL Order By

SQL Insert

SQL Null

SQL Update

SQL Delete

Exercise:

Select all the *different* values from the `Country` column in the `Customers` table.

Correct!

[Next >](#)

[Next Exercise >](#)

```
SELECT DISTINCT Country FROM Customers;
```

SQL Where

Ex01

SQL Select ✓

SQL Where

✓ Exercise 1

Exercise 2

Exercise 3

Exercise 4

Exercise 5

[Go to SQL Where Tutorial](#)

SQL Order By

SQL Insert

SQL Null

Exercise:

Select all records where the `City` column has the value "Berlin".

Correct!

[Next >](#)

[Next Exercise >](#)

```
SELECT * FROM Customers  
WHERE City = "Berlin";
```

Ex02

SQL Select	✓
SQL Where	
✓ Exercise 1	
✓ Exercise 2	
Exercise 3	
Exercise 4	
Exercise 5	
Go to SQL Where Tutorial	
SQL Order By	
SQL Insert	
SQL Null	

Exercise:

Use the `NOT` keyword to select all records where `City` is NOT "Berlin".

```
SELECT * FROM Customers
WHERE NOT City = 'Berlin';
```

Correct!

[Next >](#)

[Next Exercise >](#)

Ex03

SQL Select	✓
SQL Where	
✓ Exercise 1	
✓ Exercise 2	
✓ Exercise 3	
Exercise 4	
Exercise 5	
Go to SQL Where Tutorial	
SQL Order By	
SQL Insert	
SQL Null	

Exercise:

Select all records where the `CustomerID` column has the value 32.

```
SELECT * FROM Customers
WHERE CustomerID = 32;
```

Correct!

[Next >](#)

[Next Exercise >](#)

Ex04

SQL Select	✓
SQL Where	
✓ Exercise 1	
✓ Exercise 2	
✓ Exercise 3	
✓ Exercise 4	
Exercise 5	
Go to SQL Where Tutorial	
SQL Order By	
SQL Insert	
SQL Null	

Exercise:

Select all records where the `City` column has the value 'Berlin' and the `PostalCode` column has the value 12209.

```
SELECT * FROM Customers
WHERE City = 'Berlin'
AND PostalCode = 12209;
```

Correct!

[Next >](#)

[Next Exercise >](#)

Ex05

Completed 8 of 52 Exercises:

- SQL Select ✓
- SQL Where ✓
- ✓ Exercise 1
- ✓ Exercise 2
- ✓ Exercise 3
- ✓ Exercise 4
- ✓ Exercise 5
- [Go to SQL Where Tutorial](#)
- SQL Order By
- SQL Insert
- SQL Null

Exercise:

Select all records where the `City` column has the value 'Berlin' or 'London'.

Correct!

[Next >](#)

[Next Exercise >](#)

```
SELECT * FROM Customers
WHERE City = 'Berlin'
OR City = 'London';
```

SQL Order By

Ex01

Completed 9 of 52 Exercises:

- SQL Select ✓
- SQL Where ✓
- SQL Order By
- ✓ Exercise 1
- Exercise 2
- Exercise 3
- [Go to SQL Order By Tutorial](#)
- SQL Insert
- SQL Null
- SQL Update
- SQL Delete

Exercise:

Select all records from the `Customers` table, sort the result alphabetically by the column `City`.

Correct!

[Next >](#)

[Next Exercise >](#)

```
SELECT * FROM Customers
ORDER BY City;
```

Ex02

Completed 10 of 52 Exercises:

- SQL Select ✓
- SQL Where ✓
- SQL Order By
- ✓ Exercise 1
- ✓ Exercise 2
- Exercise 3
- [Go to SQL Order By Tutorial](#)
- SQL Insert
- SQL Null
- SQL Update
- SQL Delete

Exercise:

Select all records from the `Customers` table, sort the result reversed alphabetically by the column `City`.

Correct!

[Next >](#)

[Next Exercise >](#)

```
SELECT * FROM Customers
ORDER BY City DESC;
```

Ex03

Completed 11 of 52 Exercises:

- SQL Select ✓
- SQL Where ✓
- SQL Order By ✓
- ✓ Exercise 1
- ✓ Exercise 2
- ✓ Exercise 3
- [Go to SQL Order By Tutorial](#)
- SQL Insert
- SQL Null
- SQL Update

Exercise:

Select all records from the Customers table, sort the result alphabetically, first by the column Country, then, by the column City.

Correct!

[Next >](#)

[Next Exercise >](#)

```
SELECT * FROM Customers
ORDER BY Country, City;
```

SQL Insert

Ex1

Completed 12 of 52 Exercises:

- SQL Select ✓
- SQL Where ✓
- SQL Order By ✓
- SQL Insert ✓
- ✓ Exercise 1
- [Go to SQL Insert Tutorial](#)
- SQL Null
- SQL Update
- SQL Delete
- SQL Functions
- SQL Like
- SQL Wildcards
- SQL In
- SQL Between
- SQL Alias
- SQL Join

Exercise:

Insert a new record in the Customers table.

Correct!

[Next >](#)

[Next Exercise >](#)

```
INSERT INTO Customers (
  CustomerName,
  Address,
  City,
  PostalCode,
  Country)
VALUES (
  'Hekkan Burger',
  'Gateveien 15',
  'Sandnes',
  '4306',
  'Norway');
```

SQL Null

Ex01

Completed 13 of 52 Exercises:

- SQL Select ✓
- SQL Where ✓
- SQL Order By ✓
- SQL Insert ✓
- SQL Null
- ✓ Exercise 1
- Exercise 2
- [Go to SQL Null Tutorial](#)
- SQL Update
- SQL Delete
- SQL Functions

Exercise:

Select all records from the `Customers` where the `PostalCode` column is empty.

Correct!

[Next >](#)

[Next Exercise >](#)

```
SELECT * FROM Customers
WHERE PostalCode IS NULL;
```

Ex02

Completed 14 of 52 Exercises:

- SQL Select ✓
- SQL Where ✓
- SQL Order By ✓
- SQL Insert ✓
- SQL Null ✓
- ✓ Exercise 1
- ✓ Exercise 2
- [Go to SQL Null Tutorial](#)
- SQL Update
- SQL Delete
- SQL Functions

Exercise:

Select all records from the `Customers` where the `PostalCode` column is NOT empty.

Correct!

[Next >](#)

[Next Exercise >](#)

```
SELECT * FROM Customers
WHERE PostalCode IS NOT NULL;
```

SQL Update

Ex01

Completed 15 of 52 Exercises:

- SQL Select ✓
- SQL Where ✓
- SQL Order By ✓
- SQL Insert ✓
- SQL Null ✓
- SQL Update
- ✓ Exercise 1
- Exercise 2
- Exercise 3

Exercise:

Update the `City` column of all records in the `Customers` table.

Correct!

[Next >](#)

```
UPDATE Customers
SET City = 'Oslo';
```

Ex02

Completed 16 of 52 Exercises:

- SQL Select ✓
- SQL Where ✓
- SQL Order By ✓
- SQL Insert ✓
- SQL Null ✓
- SQL Update ✓
- ✓ Exercise 1
- ✓ Exercise 2
- Exercise 3
- [Go to SQL Update Tutorial](#)

Exercise:

Set the value of the 'City' columns to 'Oslo', but only the ones where the 'Country' column has the value 'Norway'.

Correct!

[Next >](#)

[Next Exercise >](#)

```
UPDATE Customers
SET City = 'Oslo'
WHERE Country = 'Norway';
```

Ex03

Completed 17 of 52 Exercises:

- SQL Select ✓
- SQL Where ✓
- SQL Order By ✓
- SQL Insert ✓
- SQL Null ✓
- SQL Update ✓
- ✓ Exercise 1
- ✓ Exercise 2
- ✓ Exercise 3

Exercise:

Update the 'City' value *and* the 'Country' value.

Correct!

[Next >](#)

```
UPDATE Customers
SET City = 'Oslo',
    Country = 'Norway'
WHERE CustomerID = 32;
```

SQL Delete

Ex01

Completed 18 of 52 Exercises:

- SQL Select ✓
- SQL Where ✓
- SQL Order By ✓
- SQL Insert ✓
- SQL Null ✓
- SQL Update ✓
- SQL Delete ✓
- ✓ Exercise 1
- Exercise 2
- [Go to SQL Delete Tutorial](#)

Exercise:

Delete all the records from the 'Customers' table where the 'Country' value is 'Norway'.

Correct!

[Next >](#)

[Next Exercise >](#)

```
DELETE FROM Customers
WHERE Country = 'Norway';
```

Ex02

Completed 19 of 52 Exercises:

- SQL Select ✓
- SQL Where ✓
- SQL Order By ✓
- SQL Insert ✓
- SQL Null ✓
- SQL Update ✓
- SQL Delete ✓
- ✓ Exercise 1
- ✓ Exercise 2

Exercise:

Delete all the records from the `Customers` table.

Correct!

[Next >](#)

```
DELETE FROM Customers;
```

SQL Functions

Ex01

Completed 20 of 52 Exercises:

- SQL Select ✓
- SQL Where ✓
- SQL Order By ✓
- SQL Insert ✓
- SQL Null ✓
- SQL Update ✓
- SQL Delete ✓
- SQL Functions
- ✓ Exercise 1
- Exercise 2

Exercise:

Use the `MIN` function to select the record with the smallest value of the `Price` column.

Correct!

[Next >](#)

[Next Exercise >](#)

```
SELECT MIN(Price)
FROM Products;
```

Ex02

Completed 21 of 52 Exercises:

- SQL Select ✓
- SQL Where ✓
- SQL Order By ✓
- SQL Insert ✓
- SQL Null ✓
- SQL Update ✓
- SQL Delete ✓
- SQL Functions
- ✓ Exercise 1
- ✓ Exercise 2

Exercise:

Use an SQL function to select the record with the highest value of the `Price` column.

Correct!

[Next >](#)

[Next Exercise >](#)

```
SELECT MAX(PRICE)
FROM Products;
```


Ex03

Completed 22 of 52 Exercises:	
SQL Select	✓
SQL Where	✓
SQL Order By	✓
SQL Insert	✓
SQL Null	✓
SQL Update	✓
SQL Delete	✓
SQL Functions	
✓ Exercise 1	
✓ Exercise 2	
✓ Exercise 3	

Exercise:

Use the correct function to return the number of records that have the `Price` value set to 18.

Correct!

[Next >](#)

[Next Exercise >](#)

```
SELECT COUNT(*)
FROM Products
WHERE Price = 18;
```

Ex04

Completed 23 of 52 Exercises:	
SQL Select	✓
SQL Where	✓
SQL Order By	✓
SQL Insert	✓
SQL Null	✓
SQL Update	✓
SQL Delete	✓
SQL Functions	
✓ Exercise 1	
✓ Exercise 2	
✓ Exercise 3	
✓ Exercise 4	

Exercise:

Use an SQL function to calculate the average price of all products.

Correct!

[Next >](#)

[Next Exercise >](#)

```
SELECT AVG(Price)
FROM Products;
```

Ex05

Completed 24 of 52 Exercises:	
SQL Select	✓
SQL Where	✓
SQL Order By	✓
SQL Insert	✓
SQL Null	✓
SQL Update	✓
SQL Delete	✓
SQL Functions	✓
✓ Exercise 1	
✓ Exercise 2	
✓ Exercise 3	
✓ Exercise 4	
✓ Exercise 5	

Exercise:

Use an SQL function to calculate the sum of all the `Price` column values in the `Products` table.

Correct!

[Next >](#)

[Next Exercise >](#)

```
SELECT SUM(Price)
FROM Products;
```