The OFFSET and FETCH clauses are the options of the [ORDER BY](https://www.sqlservertutorial.net/sql-server-basics/sql-server-order-by/) clause. They allow you to limit the number of rows to be returned by a [query](https://www.sqlservertutorial.net/sql-server-basics/sql-server-select/).

The following illustrates the syntax of the OFFSET and FETCH clauses:

ORDER BY column\_list [ASC |DESC]

OFFSET offset\_row\_count {ROW | ROWS}

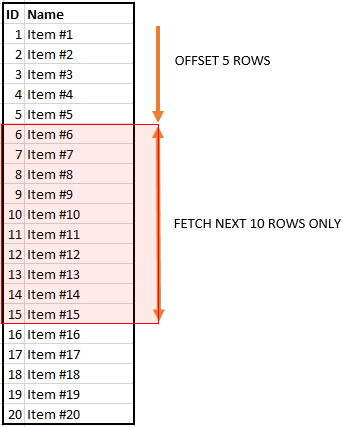
FETCH {FIRST | NEXT} fetch\_row\_count {ROW | ROWS} ONLY

Code language: SQL (Structured Query Language) (sql)

In this syntax:

* The OFFSET clause specifies the number of rows to skip before starting to return rows from the query. The offset\_row\_count can be a constant, variable, or parameter that is greater or equal to zero.
* The FETCH clause specifies the number of rows to return after the OFFSET clause has been processed. The offset\_row\_count can a constant, variable or scalar that is greater or equal to one.
* The OFFSET clause is mandatory while the FETCH clause is optional. Also, the FIRST and NEXT are synonyms respectively so you can use them interchangeably. Similarly, you can use the FIRST and NEXT interchangeably.

The following illustrates the OFFSET and FETCH clauses:



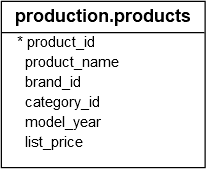
Note that you must use the OFFSET and FETCH clauses with the ORDER BY clause. Otherwise, you will get an error.

The OFFSET and FETCH clauses are preferable for implementing the query paging solution than the TOP clause.

The OFFSET and FETCH clauses have been available since SQL Server 2012 (11.x) and later and Azure SQL Database.

SQL Server OFFSET and FETCH examples

We will use the products table from the [sample database](https://www.sqlservertutorial.net/sql-server-sample-database/) for the demonstration.



The following query returns all products from the products table and sorts the products by their list prices and names:

SELECT

product\_name,

list\_price

FROM

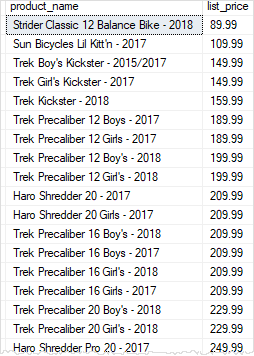
production.products

ORDER BY

list\_price,

product\_name;

Code language: SQL (Structured Query Language) (sql)



To skip the first 10 products and return the rest, you use the OFFSET clause as shown in the following statement:

SELECT

product\_name,

list\_price

FROM

production.products

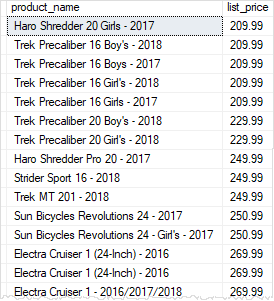
ORDER BY

list\_price,

product\_name

OFFSET 10 ROWS;

Code language: SQL (Structured Query Language) (sql)



To skip the first 10 products and select the next 10 products, you use both OFFSET and FETCH clauses as follows:

SELECT

product\_name,

list\_price

FROM

production.products

ORDER BY

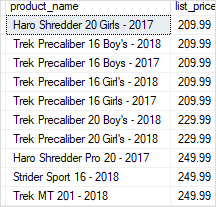
list\_price,

product\_name

OFFSET 10 ROWS

FETCH NEXT 10 ROWS ONLY;

Code language: SQL (Structured Query Language) (sql)



To get the top 10 most expensive products you use both OFFSET and FETCH clauses:

SELECT

product\_name,

list\_price

FROM

production.products

ORDER BY

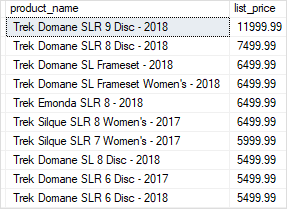
list\_price DESC,

product\_name

OFFSET 0 ROWS

FETCH FIRST 10 ROWS ONLY;

Code language: SQL (Structured Query Language) (sql)



In this example, the ORDER BY clause sorts the products by their list prices in descending order. Then, the OFFSET clause skips zero row and the FETCH clause fetches the first 10 products from the list.