

Pivot and Unpivot in SQL

In SQL, Pivot and Unpivot are relational operators that are used to transform one table into another in order to achieve more simpler view of table. Conventionally we can say that **Pivot** operator converts the rows data of the table into the column data. The **Unpivot** operator does the opposite that is it transform the column based data into rows.

Syntax

1. Pivot

```
SELECT (ColumnName)  
FROM (TableName)  
PIVOT  
(  
    AggregateFunction(ColumnToBeAggregated)  
    FOR PivotColumn  
    IN (PivotColumnValues)  
) AS (Alias)  
// Alias is a temporary name for a table
```

2. Unpivot

```
SELECT (ColumnName)  
FROM (TableName)  
UNPIVOT  
(  
    AggregateFunction(ColumnToBeAggregated)  
    FOR PivotColumn  
    IN (PivotColumnValues)  
) AS (Alias)
```

Example

Example - 1

We have created a simple table named “geeksforgeeks” with values like Course name, course category and price and inserted the respective values.

```
CREATE TABLE geeksforgeeks  
(  
    CourseName nvarchar(50),  
    CourseCategory nvarchar(50),  
    Price int  
);  
INSERT INTO geeksforgeeks VALUES  
( 'C', 'PROGRAMMING', 5000)  
( 'JAVA', 'PROGRAMMING', 6000)
```

```
( 'PYTHON', 'PROGRAMMING', 8000)
( 'PLACEMENT 100', 'INTERVIEWPREPARATION', 5000);
SELECT *
FROM geeksforgeeks;
```

The output we get is:

CourseName	CourseCategory	Price
C	PROGRAMMING	5000
JAVA	PROGRAMMING	6000
PYTHON	PROGRAMMING	8000
PLACEMENT 100	INTERVIEWPREPARATION	5000

Now, applying PIVOT operator to this data:

```
SELECT CourseName, PROGRAMMING, INTERVIEWPREPARATION
FROM geeksforgeeks
PIVOT
(
    SUM(Price)
    FOR CourseCategory
    IN
        (PROGRAMMING, INTERVIEWPREPARATION)
) AS PivotTable
```

After using Pivot operator we get the following result:

CourseName	PROGRAMMING	InterviewPreparation
C	5000	NULL
JAVA	6000	NULL
PLACEMENT 100	NULL	5000
PYTHON	8000	NULL

Example - 2

Now, we use the same table “geeksforgeeks” created in the above example and apply the Unpivot operator to our Pivoted table.

Applying UNPIVOT operator:

```
SELECT CourseName, CourseCategory, Price
FROM
(
    SELECT CourseName,
```

```

PROGRAMMING,
INTERVIEWPREPARATION
FROM geeksforgeeks
PIVOT
(
    SUM(Price)
    FOR CourseCategory
    IN
    (PROGRAMMING, INTERVIEWPREPARATION)
)
AS PivotTable
) P
UNPIVOT
(
    Price
    FOR CourseCategory
    IN
    (PROGRAMMING, INTERVIEWPREPARATION)
)
AS UnpivotTable

```

After using Unpivot operator we get our original table back as we have successfully transformed the columns of the table back to rows:

CourseName	CourseCategory	Price
C	PROGRAMMING	5000
JAVA	PROGRAMMING	6000
PLACEMENT 100	INTERVIEWPREPARATION	5000
PYTHON	PROGRAMMING	8000