

# How do you create tables in Python? Can you provide some examples?

Authored by  
**stats writer**

May 5, 2024

## RECOMMENDED CITATION

stats writer (2024). *How do you create tables in Python? Can you provide some examples?*. PSYCHOLOGICAL SCALES. Retrieved from <https://scales.arabpsychology.com/?p=143100>

Creating tables in Python is a process of organizing data in a structured format, with rows and columns, to store and manipulate large amounts of information efficiently. This can be achieved by using the built-in "Table" module or the popular "Pandas" library.

To create a table using the "Table" module, you first need to import it into your code. Then, using the "Table()" function, you can define the columns and rows of the table by specifying the data types and labels. Once the table is created, you can add data to it by using the "add\_row()" function.

An example of creating a table using the "Table" module would be:

```
import Table

my_table = Table.Table(, )
my_table.add_row()
my_table.add_row()
my_table.add_row()
```

To create a table using the "Pandas" library, you need to import it into your code and use the "DataFrame()" function. This function allows you to specify the data and column names for the table. You can then add data to the table using the "append()" function.

An example of creating a table using the "Pandas" library would be:

```
import pandas as pd

my_table = pd.DataFrame(
{'Name': ,
'Age': ,
'Occupation': }
)
```

Both methods allow you to create tables in Python and manipulate the data within them. This can be useful for data analysis, data visualization, and other tasks that require organized data.

## Create Tables in Python (With Examples)

**The easiest way to create tables in Python is to use `tabulate()` function from the library.**

**To use this function, we must first install the library using pip:**

```
pip install tabulate
```

**We can then load the library:**

```
from tabulate import tabulate
```

**We can then use the following basic syntax to create tables:**

```
print(tabulate(data, headers=col_names, tablefmt="grid", showindex="always"))
```

**The following examples show how to use this function in practice.**

**Example 1: Create Table with Headers**

**The following code shows how to create a basic table with headers:**

```
#create data
```

```
data = ,
```

```
,
```

```
,  
]
```

```
#define header names
```

```
col_names =
```

```
#display table
```

```
print(tabulate(data, headers=col_names))
```

```
Team Points
```

```
-----
```

```
Mavs 99
```

```
Suns 91
```

```
Spurs 94
```

```
Nets 88
```

```
Example 2: Create Table with Fancy Grid
```

**The following code shows how to create a table with headers and a fancy grid:**

```
#create data
```

```
data = ,
```

```
,
```

```
,
```

]

**#define header names****col\_names =****#display table**

```
print(tabulate(data, headers=col_names,
tablefmt="fancy_grid"))
```

????????????????????

**? Team ? Points ?**

????????????????????

**? Mavs ? 99 ?**

????????????????????

**? Suns ? 91 ?**

????????????????????

**? Spurs ? 94 ?**

????????????????????

**? Nets ? 88 ?**

????????????????????

**Note that the tablefmt argument accepts several different options including:**

**gridfancy\_gridpipeprettysimple**

Refer to the for a complete list of potential table formats.

Example 3: Create Table with Index Column

The following code shows how to create a table with headers, a fancy grid, and an index column:

```

#create data
data = ,
,
,
]

#define header names
col_names =

#display table
print(tabulate(data, headers=col_names,
tablefmt="fancy_grid", showindex="always"))

????????????????????????????????????????
? ? Team ? Points ?
????????????????????????????????????????
? 0 ? Mavs ? 99 ?
????????????????????????????????????????

```

**? 1 ? Suns ? 91 ?**

????????????????????????????????????

**? 2 ? Spurs ? 94 ?**

????????????????????????????????????

**? 3 ? Nets ? 88 ?**

????????????????????????????????????

ARABPSYCHOLOGY.COM