

## How to create tables in pdf with pdf sdk in C# and ByteScout PDF Suite

Learning is essential in computer world and the tutorial below will demonstrate how to create tables in pdf with pdf sdk in C#

This sample source code below will display you how to create tables in pdf with pdf sdk in C#. What is ByteScout PDF Suite? It is the set that includes 6 SDK products to work with PDF from generating rich PDF reports to extracting data from PDF documents and converting them to HTML. This bundle includes PDF (Generator) SDK, PDF Renderer SDK, PDF Extractor SDK, PDF to HTML SDK, PDF Viewer SDK and PDF Generator SDK for Javascript. It can help you to create tables in pdf with pdf sdk in your C# application.

These C# code samples for C# guide developers to speed up coding of the application when using ByteScout PDF Suite. This C# sample code is all you need for your app. Just copy and paste the code, add references (if needs to) and you are all set! Check C# sample code samples to see if they respond to your needs and requirements for the project.

You can download free trial version of ByteScout PDF Suite from our website to see and try many others source code samples for C#.

FOR MORE INFORMATION AND FREE TRIAL:

[Download Free Trial SDK \(on-premise version\)](#)

[Read more about ByteScout PDF Suite](#)

[Explore API Documentation](#)

[Get Free Training for ByteScout PDF Suite](#)

[Get Free API key for Web API](#)

[visit www.Bytescout.com](http://www.Bytescout.com)

Source Code Files:

```
using System;
using System.Diagnostics;
using Bytescout.PDF;

namespace Tables
{
    /// <summary>
    /// This example demonstrates how to create tables.
    /// </summary>
    class Program
    {
        static void Main()
        {
            // Create new document
            Document pdfDocument = new Document();
            pdfDocument.RegistrationName = "demo";
            pdfDocument.RegistrationKey = "demo";
            // Add page
            Page page = new Page(PaperFormat.A4);
            pdfDocument.Pages.Add(page);

            DeviceColor lightGrayColor = new ColorGray(200);
            DeviceColor whiteColor = new ColorGray(255);
            DeviceColor lightBlueColor = new ColorRGB(200, 200, 250);
            DeviceColor lightRedColor = new ColorRGB(255, 200, 200);

            // Create a table and set default background color
            Table table = new Table();
            table.BackgroundColor = lightGrayColor;

            // Add row headers column and set its color
            table.Columns.Add(new TableColumn("RowHeaders"));
            table.Columns[0].BackgroundColor = lightGrayColor;

            // Add columns A, B, C, ...
            for (int c = 0; c < 10; c++)
            {
                string columnName = Convert.ToChar('A' + c).ToString();
                table.Columns.Add(new TableColumn(columnName, columnName));
            }

            // Add rows
            for (int r = 0; r < 10; r++)
            {
                // Create new row and set its background color
                TableRow row = table.NewRow();
                row.BackgroundColor = whiteColor;

                // Set row header text
                row["RowHeaders"].Text = (r + 1).ToString();

                // Set cell text
                for (int c = 0; c < 10; c++)
                {
```



[Get Your API Key](#)  
[Explore Web API Docs](#)  
[Explore Web API Samples](#)

[visit www.ByteScout.com](http://www.ByteScout.com)

[visit www.PDF.co](http://www.PDF.co)

[www.bytescout.com](http://www.bytescout.com)