

How to use text auto corrections in PowerShell and ByteScout Text Recognition SDK

Write code in PowerShell to use text auto corrections with this step-by-step tutorial

The sample source code below will teach you how to use text auto corrections in PowerShell. ByteScout Text Recognition SDK: the text recognition SDK to help with extraction of text using OCR from scanned images and documents. Supports English and non-Latin languages, can take PDF as input. It can use text auto corrections in PowerShell.

Fast application programming interfaces of ByteScout Text Recognition SDK for PowerShell plus the instruction and the code below will help you quickly learn how to use text auto corrections. This PowerShell 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! Enjoy writing a code with ready-to-use sample PowerShell codes.

You can download free trial version of ByteScout Text Recognition SDK from our website to see and try many others source code samples for PowerShell.

PowerShell - AutoCorrections.ps1

```
# Add reference to ByteScout.TextRecognition.dll assembly
Add-Type -Path "c:\Program Files\ByteScout Text Recognition
SDK\net40\ByteScout.TextRecognition.dll"

$InputDocument = "..\..\bad-quality.png"
$OutputDocument = ".\result.txt"

# Create and activate TextRecognizer instance
$textRecognizer = New-Object ByteScout.TextRecognition.TextRecognizer
$textRecognizer.RegistrationName = "demo"
$textRecognizer.RegistrationKey = "demo"

try {
    # Load document (image or PDF)
    $textRecognizer.LoadDocument($InputDocument)

    # Set the location of OCR language data files
    $textRecognizer.OCRLanguageDataFolder = "c:\Program Files\ByteScout Text
Recognition SDK\ocrdata_fast\"

    # Set OCR language.
    # "eng" for english, "deu" for German, "fra" for French, "spa" for Spanish, etc.
    - according to files in "ocrdata" folder
    # Find more language files at https://github.com/bytescout/ocrdata
    $textRecognizer.OCRLanguage = "eng"
```

```
# Add error corrections that will be applied after the recognition.
$textRecognizer.Corrections.Add("Tut ", "Test ")
$textRecognizer.Corrections.Add("Recog\\w{1,}on", "Recognition", $true) # regular
expression (regex) replacement

# Recognize text from all pages and save it to file
$textRecognizer.SaveText($OutputDocument)

# Open the result file in default associated application (for demo purposes)
& $OutputDocument
}
catch {
# Display exception
Write-Host $_.Exception.Message
}

$textRecognizer.Dispose()
```

PowerShell - run.bat

```
@echo off

powershell -NoProfile -ExecutionPolicy Bypass -Command "& .\AutoCorrections.ps1"
echo Script finished with errorlevel=%errorlevel%

pause
```

FOR MORE INFORMATION AND FREE TRIAL:

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

[Read more about ByteScout Text Recognition SDK](#)

[Explore documentation](#)

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

or

[Get Your Free API Key for www.PDF.co Web API](https://www.pdf.co/web-api)