



maven-central v9.8.0 license AGPL downloads 142k commit activity 36/month

iText Core/Community is a high-performance, battle-tested library that allows you to create, adapt, inspect and maintain PDF documents, allowing you to add PDF functionality to your software projects with ease. It is also available for [.NET](#) (C#).

The key features of iText Core/Community are:

- Core library:
 - PDF creation with the use of our layout engine
 - PDF manipulation, e.g. merging multiple PDFs into one, adding new content, ...
 - PDF digital signing
 - PDF form creation and manipulation
 - Working with PDF/A documents
 - Working with PDF/UA documents
 - FIPS-compliant cryptography
 - Barcode generation
 - SVG support
- [Addons](#):
 - Converting XML/HTML & CSS to PDF [repo](#), [info](#)
 - Perform OCR on images and PDF documents [repo](#), [info](#)
 - Redacting sensitive information in PDF documents [repo](#), [info](#)
 - Support for international character sets (e.g. Arabic, Chinese, Hebrew, Thai, ...) [info](#)
 - Optimize PDF documents for reduced file size, and increased performance [info](#)
 - Flattening XFA documents [info](#)
 - PDF debugging [repo](#), [info](#)

Want to discover what's possible? Head over to our [Demo Lab](#)! It contains a collection of demo applications ready to use online!

Getting started

The easiest way to get started is to use Maven, just add the following entries to your pom.xml file:

```

<properties>
  <itext.version>REPLACE_WITH_DESIRED_ITEXT_VERSION</itext.version>
</properties>
<dependencies>
  <dependency>
    <groupId>com.itextpdf</groupId>
    <artifactId>itext-core</artifactId>
    <version>${itext.version}</version>
    <type>pom</type>
  </dependency>
  <dependency>
    <groupId>com.itextpdf</groupId>
    <artifactId>bouncy-castle-adapter</artifactId>
    <version>${itext.version}</version>
  </dependency>
</dependencies>

```

For more advanced use cases, please refer to the [Installation guidelines](#). You can also [build iText Community from source](#).

Hello PDF!

The following example shows how easy it is to create a simple PDF document:

```

package com.itextpdf.hellopdf;

import com.itextpdf.kernel.pdf.PdfDocument;
import com.itextpdf.kernel.pdf.PdfWriter;
import com.itextpdf.layout.Document;
import com.itextpdf.layout.element.Paragraph;

import java.io.FileNotFoundException;

public class HelloPdfApp {

    public static void main(String[] args) throws FileNotFoundException {
        try (Document document = new Document(new PdfDocument(new
PdfWriter("./hello-pdf.pdf")))) {
            document.add(new Paragraph("Hello PDF!"));
        }
    }
}

```

GraalVM compatibility

Starting from version 8.0.3 iText Core supports native image compilation using GraalVM. See [building](#) for details.

Examples

This is a small subset of examples to get you started. For more advanced examples, refer to our [Knowledge Base](#) or the following links: [Examples repo](#), [Signing examples](#).

Some of the output PDF files will be incorrectly displayed by the GitHub previewer, so be sure to download them to see the correct results.

Description	Link
Basic layout	
Change text properties	Java , PDF
Creating a simple table	Java , PDF
Add an image to a PDF document	Java , PDF
Create a list	Java , PDF
Add a watermark	Java , PDF
Add links to navigate within a document	Java , PDF
Create a popup annotation	Java , PDF
Change font	Java
Add form fields	Java
General document settings	
Change page size and margin	Java , PDF
Write PDF to byte array instead of to disk	Java
Change page rotation	Java , PDF
Add header and footer	Java , PDF
Add dynamic page margins	Java , PDF
Add footnotes	Java , PDF
Merge documents	Java , PDF
Flatten annotations	Java

Description	Link
PDF/UA, PDF/A	
Create PDF/UA document	Java , PDF
Create PDF/A-3 document	Java
Create PDF/A-4 document	Java
Create WTPDF document	Java
Create ZUGFeRD/Factur-X document	Java , PDF
Digital signatures	
Sign PDF document	Java , PDF
Customize signature appearance	Java , PDFs
Sign PDF document with PaDES Profile	Java , PDFs
Two-phase signing	Java , PDFs
Signature validation	Java
Signing using PQC algorithms (experimental)	Java , PDFs
Custom HTML tag for signature	Java , HTML , PDF
FIPS	
Enable FIPS	Java
FIPS SHA3 example	Java
SVG	
Convert SVG to a PDF	Java
Convert SVG to a PDF using layout	Java
Convert SVG as string to PDF	Java
Convert SVG to a PdfPage	Java

Description	Link
Convert SVG as XObject	Java
Convert SVG to a PDF with pdfCalligraph	Java
Convert HTML and CSS to PDF	Link to repo
Convert simple HTML doc to PDF	Java
OCR images or PDFs	Link to repo
OCR images and create PDF	Using ONNX models , using Tesseract 4
Make PDF searchable	Using ONNX models , using Tesseract 4
OCR images to txt file	Using ONNX models , using Tesseract 4
Secure redaction of content	Link to repo
Redacting content	Java
Redact based on regex	Java
Support complex writing systems	Link to docs
Add Arabic text	Java , PDF
Optimizing PDFs	Link to docs
Reduce size of PDF	Java
XFA flattening	Link to docs
Flatten an XFA document	Java
RUPS	Link to repo

Description	Link
Debug a PDF	Latest release

FAQs, tutorials, etc.

Check out the [iText Knowledge Base](#) for the [iText Jump-start tutorial](#) and other tutorials, [FAQs](#) and more. For specific information and examples relating to digital signatures and iText, make sure to check the [Digital Signatures Hub](#).

Many common questions have already been answered on [Stack Overflow](#), so make sure to also check there.

Contributing

Many people have contributed to **iText Core/Community** over the years. If you've found a bug, a mistake in documentation, or have a hot new feature you want to implement, we welcome your contributions.

Small changes or fixes can be submitted as a [Pull Request](#), while for major changes we request you contact us at community@apryse.com so we can better coordinate our efforts and prevent duplication of work.

Please read our [Contribution Guidelines](#) for details on code submissions, coding rules, and more.

Licensing

iText is dual licensed as [AGPL/Commercial software](#).

AGPL is a free/open-source software license, however, this doesn't mean the software is [gratis](#)!

The AGPL is a copyleft license, which means that any derivative work must also be licensed under the same terms. If you're using iText in software or a service which cannot comply with the AGPL terms, we have a commercial license available that exempts you from such obligations.

Contact [Sales](#) for more info.

Markdown to PDF

Along with the release of iText Core 9.8.0, we also launched a free [GitHub Action](#) for open-source developers to easily convert any .md file to a **fully-accessible PDF/UA-2** document.