K Open Pdf Showcase

Simple Kotlin DSL for creating simple Pdf document

1. Document properties (metadata)

A document properties shows basic information about the document. The title, author, subject, and keywords may have been set by the person who created the document in the source application.

> See also PDF properties and metadata

Code: Set the document properties

```
documentInfo {
  title = "Title is here"
  author = "Author is here"
  subject = "Subject is here"
  keywords = listOf("key1", "key2", ...)
  creator = "Creator Info"
}
```

2. Viewing PDFs and viewing preferences

The initial view of the PDF depends on how its creator set the document properties. For example, a document may open at a particular page or magnification.

> See also Viewing Preferences

Code: Set the view preferences

```
documentViewPreferences {
  pageLayout = PageLayout.OneColumn
  pageMode = PageMode.UseNone
  fitWindow = true
}
```

3. Overview of security in Acrobat and PDFs

Security applies in two general contexts: application (software) security and content security.

Application security involves customizing security features to protect Acrobat and Reader against vulnerabilities, malicious attacks, and other risks. Advanced users can customize the application through the user interface. Enterprise administrators can also configure the registry. See the following articles for details.

Content security involves using product features to protect the integrity of PDF content. These features safeguard against the unwanted alteration of PDFs, keep sensitive information private, prevent the printing of PDFs, and so on. See the following articles for details.

> See also Security in PDF

Code: Set document encryption

```
documentEncryption {
  encryptionType = EncryptionType.STANDARD_ENCRYPTION_128
  setOwnerPassword("password".toByteArray(Charsets.UTF_8))

  allowPrinting = true
  allowModifyAnnotations = true
  allowFillIn = true
  allowAssembly = true
}
```

4. Watermark

A **watermark** is an identifying image or pattern in paper that appears as various shades of lightness/darkness when viewed by transmitted light (or when viewed by reflected light, atop a dark background), caused by thickness or density variations in the paper. Watermarks have been used on postage stamps, currency, and other government documents to discourage counterfeiting. There are two main ways of producing watermarks in paper; the dandy roll process, and the more complex cylinder mould process.

Watermarks vary greatly in their visibility; while some are obvious on casual inspection, others require some study to pick out. Various aids have been developed, such as watermark fluid that wets the paper without damaging it. A watermark is very useful in the examination of paper because it can be used for dating documents and artworks, identifying sizes, mill trademarks and locations, and determining the quality of a sheet of paper. The word is also used for digital practices that share similarities with physical watermarks. In one case, overprint on computer-printed output may be used to identify output from an unlicensed trial version of a program. In another instance, identifying codes can be encoded as a digital watermark for a music, video, picture, or other file. Or an artist adding their identifying digital Signature, graphic, logo in their digital artworks as an identifier or anti-counterfeit measure.

> See also Watermark

Code: Add a Watermark

```
watermark("Text")
```

K Open Pdf Showcase

5. Page Header and Footer

Page header

In typography and word processing, a page header (or simply header) is text that is separated from the body text and appears at the top of a printed page. Word-processing programs usually allow for the configuration of page headers, which are typically identical throughout a work except in aspects such as page numbers.

> See also Page header

Page footer

In typography and word processing, the page footer (or simply footer) of a printed page is a section located under the main text, or body. It is typically used as the space for the page number. In the earliest printed books it also contained the first words of the next page; in this case they preferred to place the page number in the page header, in the top margin. Because of the lack of a set standard, in modern times the header and footer are sometimes interchangeable. In some instances, there are elements of the header inserted into the footer, such as the book or chapter title, the name of the author or other information. In the publishing industry the page footer is traditionally known as the running foot, whereas the page header is the running head.

> See also Page footer

Code: Add a page header and footer

```
header("<color DarkCyan>K Open Pdf Showcase</color>")
footer("Version 2024.12")
```

6. § Chapter and Section

A chapter is any of the main thematic divisions within a writing of relative length. A section is a subdivision, especially of a chapter.

> See also Chapter, Section

Code: Create a Chapter

```
body {
  chapter("Title", 1) {
    triggerNewPage()
    ...
  }
  chapter("Title", 2) {
    triggerNewPage()
    ...
  }
}
```

Code: Create a Section

```
body {
  section("Title 1") { ... }
  section("Title 2") { ... }
}
```

Code: Add a Section inside the Chapter

```
body {
  chapter("Title", ChapterType.AutoNumber) {
    section("Sub Title 1") { ... }
    section("Sub Title 2") { ... }
  }
}
```

7. ¶ Paragraph

A paragraph is a self-contained unit of discourse in writing dealing with a particular point or idea. Though not required by the orthographic conventions of any language with a writing system, paragraphs are a conventional means of organizing extended segments of prose.

> See also Paragraph

Inline Text

Code: Add a paragraph

```
body {
  paragraph("Inline text")
  paragraph {
    text("Any text")
  }
}
```

Bold
Italic
Underline
~~Strikethru~~
~~Bold + Italic + Underline + Strikethru~~

Color RED
Color GREEN
Color BLUE

Code: Use some tags inside text

```
body {
  paragraph("""
    <b>Bold</b>
    <i>Italic</i>
    <u>Underline</u>
    <s>Strikethru</s>
    <b><i><u><s>Bold + Italic + Underline + Strikethru</s></u></i></b>

    <color #FF358A>RED</color>
    <color #56A76B>GREEN</color>
    <color rgb(53, 116, 240)>BLUE</color>
  """)
}
```

Heading 1

Heading 2

Heading 3

K Open Pdf Showcase

Heading 4

Heading 5

Heading 6

Code: Add a heading

```
h1("Heading 1")
h2("Heading 2")
h3("Heading 3")
h4("Heading 4")
h5("Heading 5")
h6("Heading 6")
```

Code: Add an icon (Font Awesome 4.7.0)

```
body {

  TagPhrase.registerFAIconHandler()

  paragraph {
    text("...")
    fa('\uF087', 11f, java.awt.Color.GREEN)
    text("...")
  }
  paragraph("Icon <fa \uF087> inside text line!")
}
```

Some long text to be wrapped by
specified length
Some long text to be wrapped by
specified length and different
delimiters

Code: Use word wrap extension method

```
text("...".wordWrap(40))
text("...".wordWrapBy(40, ' ', ','))
```

K Open Pdf Showcase

8. Table

A **table** is an arrangement of information or data, typically in rows and columns, or possibly in a more complex structure. Tables are widely used in communication, research, and data analysis. Tables appear in print media, handwritten notes, computer software, architectural ornamentation, traffic signs, and many other places. The precise conventions and terminology for describing tables vary depending on the context. Further, tables differ significantly in variety, structure, flexibility, notation, representation and use. Information or data conveyed in table form is said to be in tabular format (adjective). In books and technical articles, tables are typically presented apart from the main text in numbered and captioned floating blocks.

> See also Table (information)

Simple table

No	Country	Name	Email	Phone Number	Currency
1	Antarctica	Akew	tsar.prawn@email.com	+1-202-555-1746	956
2	Vanuatu	Hazel	mellow.coral@email.com	+1-202-555-7985	104
3	North Korea	Prawn	cesar.prawn@email.com	+1-202-555-7887	764

No data table

Digital code	Letter code	Unit	Currency name	Rate
No Exchange Rates				

Transposed table

Id	ae4ad64f-4488-4a70-9889-b53580fb340e
First Name	Rock
Second Name	–
Email	cod.rock@email.com

Designed table (TableStyleLight16)

No	Id	First Name	Second Name
1	3e678b83-5454-4d4c-a41e-6d6a3d8ece34	Tsar	Mellow
2	cf62bdae-1b2e-48e2-a198-bdb7673ec417	Doe	Doe
3	699a3648-f3df-42f2-b80d-cd07d231ef8b	Hazel	Nutt

Custom Designed table

No	Id	First Name	Second Name
1	85d490e2-66d4-4724-8307-1f7e4f688ca9	Hazel	Cesar

K Open Pdf Showcase

2	f19abc64-b2f9-44a4-bfef-50c3aa7e921e	Bacon	Rock
3	a969110b-e46c-4c55-9d75-92c49d756df9	Creem	Tsar

Available Designers

No	Table Designer Name	Colors		
		Border	Header	Stripes
1	NoStyle	Not Found		
2	BareboneStyle			
3	TableStyleLight1			
4	TableStyleLight2			
5	TableStyleLight3			
6	TableStyleLight4			
7	TableStyleLight5			
8	TableStyleLight6			
9	TableStyleLight7			
10	TableStyleLight8			
11	TableStyleLight9			
12	TableStyleLight10			
13	TableStyleLight11			
14	TableStyleLight12			
15	TableStyleLight13			
16	TableStyleLight14			
17	TableStyleLight15			
18	TableStyleLight16			
19	TableStyleLight17			
20	TableStyleLight18			
21	TableStyleLight19			
22	TableStyleLight20			
23	TableStyleLight21			
24	TableStyleLight3_2010			
25	TableStyleLight5_2010			
26	TableStyleLight6_2010			
27	TableStyleLight10_2010			
28	TableStyleLight12_2010			

K Open Pdf Showcase

29	TableStyleLight13_2010			
30	TableStyleLight17_2010			
31	TableStyleLight19_2010			
32	TableStyleLight20_2010			

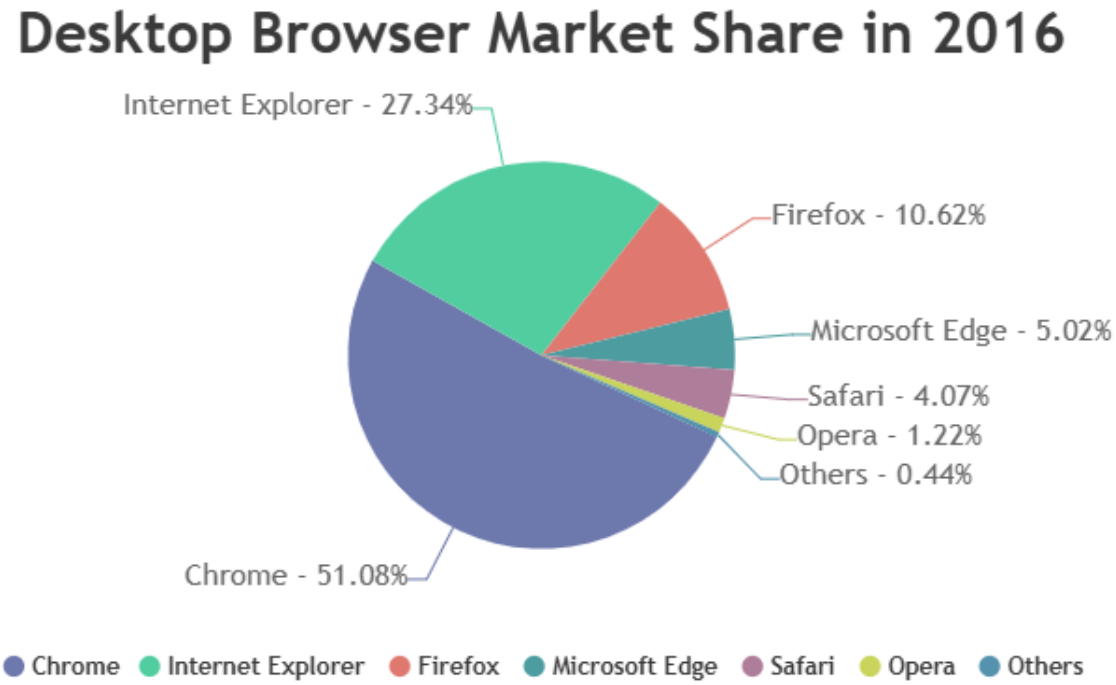
Code: Add a table

```
table(3) {
  useAllAvailableWidth()
  designer("BareboneStyle")

  columnHeaders("<b>Country</b>", "<b>Name</b>", "<b>Currency</b>")
  columnWidths(40f, 40f, 20f)

  if (hasData) {
    td(country)
    td(name)
    td(currency)
  } else {
    noData("<i>No Data</i>")
  }
}
```

9. Image



Code: Add an image to document

```
...
image("chart.png") {
  scaleToPageSize(0f, 80f)
  alignCenter()
}
...
```

K Open Pdf Showcase

PNG

Portable Network Graphics (PNG, officially pronounced /p/ PING, colloquially pronounced /pindi/ PEE-en-JEE) is a raster-graphics file format that supports lossless data compression. PNG was developed as an improved, non-patented replacement for Graphics Interchange Format (GIF)—unofficially, the initials PNG stood for the recursive acronym "**PNG's not GIF**".

PNG supports palette-based images (with palettes of 24-bit RGB or 32-bit RGBA colors), grayscale images (with or without an alpha channel for transparency), and full-color non-palette-based RGB or RGBA images. The PNG working group designed the format for transferring images on the Internet, not for professional-quality print graphics; therefore, non-RGB color spaces such as CMYK are not supported. A PNG file contains a single image in an extensible structure of chunks, encoding the basic pixels and other information such as textual comments and integrity checks documented in RFC 2083.

PNG files have the ".png" file extension and the "image/png" MIME media type. PNG was published as an informational RFC 2083 in March 1997 and as an ISO/IEC 15948 standard in 2004.

History and development

The motivation for creating the PNG format was the announcement on 28 December 1994, that implementations of the Graphics Interchange Format (GIF) format would have to pay royalties to Unisys due to their patent of the Lempel–Ziv–Welch (LZW) data compression algorithm used in GIF. This led to a flurry of criticism from Usenet users. One of them was Thomas Boutell, who on 4 January 1995 posted a precursory discussion thread on the Usenet newsgroup "comp.graphics" in which he devised a plan for a free alternative to GIF. Other users in that thread put forth many propositions that would later be part of the final file format. Oliver Fromme, author of the popular JPEG viewer QPEG, proposed the PING name, eventually becoming PNG, a recursive acronym meaning PING is not GIF, and also the .png extension. Other suggestions later implemented included the deflate compression algorithm and 24-bit color support, the lack of the latter in GIF also motivating the team to create their file format. The group would become known as the PNG Development Group, and as the discussion rapidly expanded, it later used a mailing list associated with a CompuServe forum.

The full specification of PNG was released under the approval of W3C on 1 October 1996, and later as RFC 2083 on 15 January 1997. The specification was revised on 31 December 1998 as version 1.1, which addressed technical problems for gamma and color correction. Version 1.2, released on 11 August 1999, added the iTXt chunk as the specification's only change, and a reformatted version of 1.2 was released as a second edition of the W3C standard on 10 November 2003, and as an International Standard (ISO/IEC 15948:2004) on 3 March 2004.

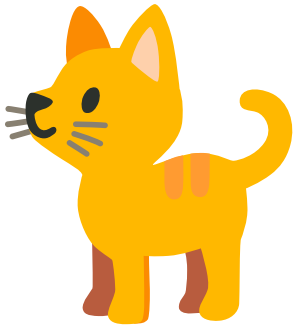
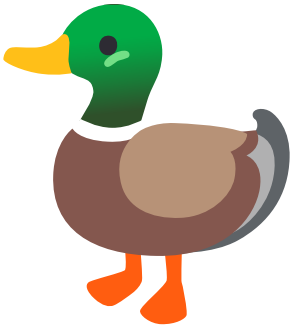
Although GIF allows for animation, it was initially decided that PNG should be a single-image format. In 2001, the developers of PNG published the Multiple-image Network Graphics (MNG) format, with support for animation. MNG achieved moderate application support, but not enough among mainstream web browsers and no usage among web site designers or publishers. In 2008, certain Mozilla developers published the Animated Portable Network Graphics (APNG) format with similar goals. APNG is a format that is natively supported by Gecko- and Presto-based web browsers and is also commonly used for thumbnails on Sony's PlayStation Portable system (using the normal PNG file extension). In 2017, Chromium based browsers adopted APNG support. In January 2020, Microsoft Edge became Chromium based, thus inheriting support for APNG. With this all major browsers now support APNG.

K Open Pdf Showcase

SVG

Scalable Vector Graphics (SVG) is an XML-based vector image format for defining two-dimensional graphics, having support for interactivity and animation. The SVG specification is an open standard developed by the World Wide Web Consortium since 1999.

SVG images are defined in a vector graphics format and stored in XML text files. SVG images can thus be scaled in size without loss of quality, and SVG files can be searched, indexed, scripted, and compressed. The XML text files can be created and edited with text editors or vector graphics editors, and are rendered by most web browsers. If used for images, SVG can host scripts or CSS, potentially leading to cross-site scripting attacks or other security vulnerabilities.


Code: Add a svg image to the document

```
<!-- Add a dependency -->
<dependency>
    <groupId>org.apache.xmlgraphics</groupId>
    <artifactId>batik-transcoder</artifactId>
    <version>1.18</version>
</dependency>

...
svg("cat.svg") {
  scaleToFit(100f)
}
...
```

10. List

A list is a set of discrete items of information collected and set forth in some format for utility, entertainment, or other purposes. A list may be memorialized in any number of ways, including existing only in the mind of the list-maker, but lists are frequently written down on paper, or maintained electronically. Lists are "most frequently a tool", and "one does not read but only uses a list: one looks up the relevant information in it, but usually does not need to deal with it as a whole".

> See also List

10.1. Basic Lists

1. First Item
2. Second Item
3. Third Item
   A) Sub Item 1
      - **Bold Item**
      - *Italic Item*
      - Underline Item
      - ~~Strikethru Item~~
   B) Sub Item 2
   C) Sub Item 3

10.2. Fancy Lists

I. First Item
II. Second Item
III. Third Item
   α) Alpha
   β) Beta
   γ) Gamma
IV. Fourth Item

10.3. Source Code

Code: Add a list to the document

```
list {
  li("First Item")
  li("Second Item ")
  li("Third Item")
  list(ListType.Letter) {
    postSymbol = ") "

    li("Sub-Item 1")

    list(ListType.Symbol) {
      li("<b>Item 1</b>")
      li("<i>Item 2</i>")
```

```
      li("<u>Item 3</u>")
      li("<s>Item 4</s>")
    }

    li("Sub-Item 2")
    li("Sub-Item 3")
  }
}
```

11. Blockquote

A **block quotation** (also known as a long quotation or extract) is a quotation in a written document that is set off from the main text as a paragraph, or block of text, and typically distinguished visually using indentation and a different typeface or smaller size font. This is in contrast to setting it off with quotation marks in a run-in quote. Block quotations are used for long quotations. The Chicago Manual of Style recommends using a block quotation when extracted text is 100 words or more, or approximately six to eight lines in a typical manuscript.

> See also Block quotation

11.1. Default Block Quotation

> The Default Block Quotation element

> The Block Quotation element with background color rgb(242, 242, 242)

> The Block Quotation element with font color rgb(165, 165, 165)

Code: Add a blockquote

```
...
blockquote {
  paragraph("The Default Block Quotation element")
}
...
```

11.2. Typed Block Quotation

> **Note**
> Useful information that users should know.

> **Tip**
> Helpful advice for doing things better or more easily.

> **Important**
> Key information users need to know to achieve their goal.

> **Warning**
> Urgent info that needs immediate user attention to avoid problems.

> **Caution**
> Advises about risks or negative outcomes of certain actions.

K Open Pdf Showcase

Code: Add a Typed blockquote

```
...
blockquote(BlockquoteType.Note) { }
blockquote(BlockquoteType.Tip) { }
blockquote(BlockquoteType.Important) { }
blockquote(BlockquoteType.Warning) { }
blockquote(BlockquoteType.Caution) { }
...
```

12. Barcode

A **barcode** or **bar code** is a method of representing data in a visual, machine-readable form.

12.1. Code 39

Code 39 (also known as **Alpha39**, **Code 3 of 9**, **Code 3/9**, **Type 39**, **USS Code 39**, or **USD-3**) is a variable length, discrete barcode symbology defined in ISO/IEC 16388:2007.

The Code 39 specification defines 43 characters, consisting of uppercase letters (A through Z), numeric digits (0 through 9) and a number of special characters (-, ., $, /, +, %, and space). An additional character (denoted '*') is used for both start and stop delimiters. Each character is composed of nine elements: five bars and four spaces. Three of the nine elements in each character are wide (binary value 1), and six elements are narrow (binary value 0).

> See also Code 39

Code: Add a CODE 39 barcode

```
...
barcode(Barcodes.code39("CODE 39"))
...
```

12.2. Code 128

Code 128 is a high-density linear barcode symbology defined in ISO/IEC 15417:2007. It is used for alphanumeric or numeric-only barcodes. It can encode all 128 characters of ASCII and, by use of an extension symbol (FNC4), the Latin-1 characters defined in ISO/IEC 8859-1.[citation needed] It generally results in more compact barcodes compared to other methods like Code 39, especially when the texts contain mostly digits. Code 128 was developed by the Computer Identics Corporation in 1981.

> See also Code 128

Code: Add a CODE 128 barcode

```
...
barcode(Barcodes.code128("RI 476 394 652 CH"))
...
```

12.3. International Article Number

The **International Article Number** (also known as **European Article Number** or **EAN**) is a standard describing a barcode symbology and numbering system used in global trade to identify a specific retail product type, in a specific packaging configuration, from a specific manufacturer. The standard has been subsumed in the Global Trade Item Number standard from the GS1 organization; the same numbers can be referred to as GTINs and can be encoded in other barcode symbologies, defined by GS1. EAN barcodes are used worldwide for lookup at retail point of sale, but can also be used as numbers for other purposes such as wholesale ordering or accounting. These barcodes only represent the digits 0–9, unlike some other barcode symbologies which can represent additional characters.

The most commonly used EAN standard is the thirteen-digit **EAN-13**, a superset of the original 12-digit Universal Product Code (UPC-A) standard developed in 1970 by George J. Laurer. An EAN-13 number includes a 3-digit GS1 prefix (indicating country of registration or special type of product). A prefix with a first digit of "0" indicates a 12-digit UPC-A code follows. A prefix with first two digits of "45" or "49" indicates a Japanese Article Number (JAN) follows.

The less commonly used 8-digit **EAN-8** barcode was introduced for use on small packages, where EAN-13 would be too large. 2-digit EAN-2 and 5-digit EAN-5 are supplemental barcodes, placed on the right-hand side of EAN-13 or UPC. These are generally used in periodicals, like magazines and books, to indicate the current year's issue number and in weighed products like food, to indicate the manufacturer's suggested retail price.

> See also International Article Number

EAN-8 

EAN-13 

UPC-A 

UPC-E 

Code: Add an EAN barcode

```
...
barcode(Barcodes.ean("4820000000222"))
barcode(Barcodes.ean("48212342", Barcodes.EanType.EAN8))
...
```

12.4. Interleaved 2 of 5

Interleaved 2 of 5 (ITF) is a continuous two-width barcode symbology encoding digits. It is used commercially on 135 film, for ITF-14 barcodes, and on cartons of some products, while the products inside are labeled with UPC or EAN. ITF was created by David Allais, who also invented barcodes Code 39, Code 11, Code 93, and Code 49.

> See also Interleaved 2 of 5

Code: Add an Interleaved barcode

```
...
barcode(Barcodes.interleaved("41-120007604-001"))
...
```

12.5. POSTNET

POSTNET (**Post**al **N**umeric **E**ncoding **T**echnique) is a barcode symbology used by the United States Postal Service to assist in directing mail. The ZIP Code or ZIP+4 code is encoded in half- and full-height bars. Most often, the delivery point is added, usually being the last two digits of the address or PO box number.

The barcode starts and ends with a full bar (often called a guard rail or frame bar and represented as the letter "S" in one version of the USPS TrueType Font) and has a check digit after the ZIP, ZIP+4, or delivery point. The encoding table is shown on the right.

Each individual digit is represented by a set of five bars, two of which are full bars (i.e. two-out-of-five code). The full bars represent "on" bits in a pseudo-binary code in which the places represent, from left to right: 7, 4, 2, 1, and 0. (Though in this scheme, zero is encoded as 11 in decimal, or in POSTNET "binary" as 11000.)

> See also Postal Numeric Encoding Technique

Code: Add a POSTNET barcode

```
...
barcode(Barcodes.postnet("5552357072"))
...
```

12.6. PLANET

The **Postal Alpha Numeric Encoding Technique** (PLANET) barcode was used by the United States Postal Service to identify and track pieces of mail during delivery – the Post Office's "CONFIRM" services. It was fully superseded by Intelligent Mail Barcode by January 28, 2013.

> See also Postal Alpha Numeric Encoding Technique

Code: Add a PLANET barcode

```
...
barcode(Barcodes.planet("4012345235636"))
...
```

12.7. PDF417

PDF417 is a stacked linear barcode format used in a variety of applications such as transport, identification cards, and inventory management. "PDF" stands for Portable Data File. The "417" signifies that each pattern in the code consists of 4 bars and spaces in a pattern that is 17 units (modules) long. The PDF417 symbology was invented by Dr. Ynjiun P. Wang at Symbol Technologies in 1991. It is defined in ISO 15438.

> See also PDF417

Code: Add a PDF417 barcode

```
...
barcode(Barcodes.pdf417("Any text"))
...
```

12.8. QR Code

A **QR code**, **quick-response code**, is a type of two-dimensional matrix barcode invented in 1994 by Masahiro Hara of Japanese company Denso Wave for labelling automobile parts. It features black squares on a white background with fiducial markers, readable by imaging devices like cameras, and processed using Reed–Solomon error correction until the image can be appropriately interpreted. The required data is then extracted from patterns that are present in both the horizontal and the vertical components of the QR image.

Whereas a barcode is a machine-readable optical image that contains information specific to the labeled item, the QR code contains the data for a locator, an identifier, and web-tracking. To store data efficiently, QR codes use four standardized modes of encoding:

1. numeric,
2. alphanumeric,
3. byte or binary, and
4. kanji.

Compared to standard UPC barcodes, the QR labeling system was applied beyond the automobile industry because of faster reading of the optical image and greater data-storage capacity in applications such as product tracking, item identification, time tracking, document management, and general marketing.

> See also QR code

K Open Pdf Showcase

Code: Add a QR Code

```
<!-- Add a dependency -->
<dependency>
    <groupId>com.google.zxing</groupId>
    <artifactId>core</artifactId>
    <version>3.5.3</version>
</dependency>

...
barcode(Barcodes.qrcode("https://en.wikipedia.org/wiki/QR_code"))
...
```

13. Simple Document Structure (Source Code)

```
NumberedPdfDocument(output).useDocument {

  defaultFont {
    size = 8f
    setFactory { size, color, style ->
      PdfDocumentFonts.getUnicodeFont("Noto Sans", size, color)
    }
  }

  documentInfo {
    title = "..."
    author = "..."
  }

  documentViewPreferences {
    pageMode = PageMode.UseOutlines
    fitWindow = true
  }

  header("Document Header")
  footer("Document Footer\nVersion")

  body {
    chapter("Chapter", 1) {
      paragraph {
        text("...")
        newLine()
        text("...")
      }
      image("...") {
        scaleToPageSize(40f, 40f)
      }
    }
    chapter("Chapter", 2) {
      paragraph("...")
      table(3) {
        useAllAvailableWidth()
        designer(DesignedTable::BareboneStyle.name)

        th("")
        th("")
        th("")

        noData("...")
      }
    }
  }
}
```

14. © Copyright

14.1. OpenPDF License

OpenPDF uses dual licensing: when using the library, you may choose either Mozilla Public License Version 2.0 or GNU Lesser General Public License 2.1.

The SPDX license identifier for OpenPDF licensing is `MPL-2.0 OR LGPL-2.1+`

Please see Mozilla Public License Version 2.0 and GNU Lesser General Public License 2.1

14.2. KOpenPDF License

KOpenPDF uses MIT licensing

> MIT License
>
> Copyright (c) 2025 Andrey Kolomiets
>
> Permission is hereby granted, free of charge, to any person obtaining a copyof this software and associated documentation files (the "Software"), to dealin the Software without restriction, including without limitation the rightsto use, copy, modify, merge, publish, distribute, sublicense, and/or sellcopies of the Software, and to permit persons to whom the Software isfurnished to do so, subject to the following conditions:
>
> The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.
>
> THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS ORIMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THEAUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHERLIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THESOFTWARE.