

# l.weasyprint-tsx

Write PDFs in TSX — Bun bundles your Preact components into HTML, WeasyPrint renders them as PDF.

---

## A.Requirements

- Bun  $\geq$  1.0
- WeasyPrint (pip install weasyprint)

## A.Quick start

1. bunx create-weasyprint-tsx my-doc
2. cd my-doc && bun install
3. bun run dev



## 1. Getting Started

The `create-weasyprint-tsx` scaffolder generates a ready-to-use project. After installation, two scripts are available:

Script	<code>bun run dev</code>	<code>bun run build</code>
Effect	Watch <code>src/</code> , rebuild on change, serve preview at <code>localhost:3000</code>	Build once, write <code>output.pdf</code>

### A. Project structure

```
my-doc/  
├─ src/  
│   ├─ index.tsx      # Document root – export a default Preact component  
│   └─ index.css      # Styles – must start with @import "tailwindcss"  
├─ weasyprint-tsx.config.ts  
└─ package.json
```

### B. Entry point

`src/index.tsx` must export a default component that renders a complete HTML document. WeasyPrint receives the full HTML string.

```
import { H1, H2, Page, UL, LI } from "@weasyprint-tsx/ui";
import "./index.css";

export default function Document() {
  return (
    <html>
      <head>
        <title>My Report</title>
        <link rel="stylesheet" href="index.css" />
      </head>
      <body>
        <Page>
          <H1>My Report</H1>
          <H2>Introduction</H2>
          <UL>
            <LI>Authored in TSX</LI>
            <LI>Rendered by WeasyPrint</LI>
          </UL>
        </Page>
      </body>
    </html>
  );
}
```



1.Configuration

Create `weasyprint-tsx.config.ts` at the project root to override any default:

```
import type { Config } from "@weasyprint-tsx/build";

const config: Config = {
  io: { output: "report.pdf" },
  weasyprint: {
    path: "weasyprint",
    optimize_images: true,
    dpi: 150,
  },
  dev: { port: 3000 },
};

export default config;
```

A.All config fields

Field	io.input	io.output	io.buildDir	io.media_type	weasyprint.path	weasyprint.dpi	weasyprint.optimize_images	weasyprint.pdf_form
Default	src/ index.tsx	output.pdf	.build	print	weasyprint	—	—	—
Description	TSX entry point	Output PDF path	Intermediate HTML directory	CSS media type	Path to WeasyPrint binary	Output resolution	Compress images in output	Enable PDF form fields



## 1.UI Components

`@weasyprint-tsx/ui` provides print-optimized Preact components. Import everything from the package root:

```
import { H1, H2, Page, PageBreak, UL, OL, LI, Table, Entry, BlockBox, Block, DotLine, CodeBlock, Equation}
```

### A.Page & Layout

Page	Wraps a page section (<section>). Accepts page?: string for named page rules.
PageBreak	Inserts a CSS page break. No props.
BlockBox	Multi-column flex container. Props: gap, basis, centered, align.
Block	Column inside BlockBox. Props: ratio, centered, align.

### B.Headings

H1 – H6	Render h1–h6. Props: marker (data-marker for CSS counters), color, fontSize.
ResetCounter	Hidden div that resets CSS counters. Use at the top of numbered sections.

### C.Lists

UL	Unordered list. Props: marker (bullet char), gap, pre.
OL	Ordered list. Props: start, counterType (decimal/lower-alpha/upper-roman/...), markerPre, markerPost, gap, pre.
LI	List item. Props: marker (override bullet), count (override counter).

### D.Table

Table	orientation "col" (default): first Entry is the header row. orientation "row": first cell of each Entry is the row header. Also: headerBg, cellBg, borderWidth, borderColor, headerFontSize, cellFontSize.
Entry	Data row/column. content: ComponentChild[] holds cell values. Accepts per-entry overrides: headerBg, cellBg.

### E.Inline elements

DotLine	Repeating dot fill line. Props: num, width, inline, color, lineHeight.
CodeBlock	Syntax-highlighted code (highlight.js). Props: language, code.
Equation	LaTeX via KaTeX. Props: tex, displayMode, aligned, chemical.