

HTML2PPTX - Quick Start Guide



Getting Started in 60 Seconds

1. Installation

```
cd /home/ubuntu/html2pptx-library
npm install
```

2. Basic Usage

Convert any HTML file to PowerPoint:

```
# Using the CLI tool directly
./bin/html2pptx input.html output.pptx

# Or if installed globally
npm link
html2pptx input.html output.pptx
```

3. Test with Examples

Try the included example files:

```
# Convert simple text boxes
./bin/html2pptx examples/"5 Text Boxes 16_9.html" output1.pptx

# Convert complex layout with diagrams
./bin/html2pptx examples/check.html output2.pptx

# Run all tests
npm test
```



Usage Examples

Command Line

```
# Basic conversion
html2pptx slide.html presentation.pptx

# With full paths
html2pptx /path/to/input.html /path/to/output.pptx

# Get help
html2pptx --help

# Check version
html2pptx --version
```

Programmatic API

```
const { convertHTML2PPTX } = require('./lib/html2pptx');

// Simple conversion
await convertHTML2PPTX('input.html', 'output.pptx');
```

What Gets Converted?

✓ Text Elements

- Text boxes with full styling
- Colors, fonts, sizes, alignment
- Bold, italic formatting

✓ Visual Elements

- Borders (color, width)
- Background colors
- SVG shapes and lines
- Images

✓ Layout

- Absolute positioning
- Relative positioning
- Flexbox layouts (simplified)
- Multi-column layouts



Test Results

All example files have been successfully tested:

- ✓ 5 Text Boxes 16_9.html → Simple text box layout
- ✓ check.html → Complex CAP triangle diagram

See `test/output/` directory for generated PowerPoint files.



Project Structure

```
html2pptx-library/
├── bin/
│   └── html2pptx           # CLI executable
├── lib/
│   └── html2pptx.js       # Main library
├── examples/
│   ├── 5 Text Boxes 16_9.html
│   └── check.html
├── test/
│   ├── test.js           # Test suite
│   └── output/           # Generated PPTX files
├── package.json
└── README.md
```



Tips for Best Results

1. **Use standard dimensions:** 1280x720px or 1920x1080px for slides
2. **Use absolute positioning:** For precise element placement
3. **Use hex colors:** For accurate color conversion (#RRGGBB)
4. **Keep it simple:** Complex CSS may not convert perfectly
5. **Test early:** Convert and check your output frequently



Troubleshooting

Problem: Elements are misaligned

- Use absolute positioning with explicit top/left values

Problem: Colors look wrong

- Use hex color codes instead of named colors or rgb()

Problem: Text is too small/large

- Adjust font-size in your HTML/CSS

Problem: Conversion fails

- Check HTML is valid and well-formed
- Ensure all paths are correct
- Run with DEBUG=1 for detailed errors



Next Steps

1. Read the full [README.md](#) (README.md) for detailed documentation
2. Explore the example files in `examples/`
3. Check the generated PowerPoint files in `test/output/`
4. Customize the library options in your code
5. Create your own HTML templates for conversion



Advanced Usage

Custom Options

```
const { HTML2PPTX } = require('./lib/html2pptx');


const converter = new HTML2PPTX({
  slideWidth: 10,      // inches
  slideHeight: 5.625,  // inches (16:9 ratio)
  background: { color: 'FFFFFF' }
});

await converter.convert('input.html', 'output.pptx');
```

Batch Conversion

```
const files = ['slide1.html', 'slide2.html', 'slide3.html'];

for (const file of files) {
  const output = file.replace('.html', '.pptx');
  await convertHTML2PPTX(file, output);
  console.log(`Converted: ${file} → ${output}`);
}
```

 **You're ready to go!** Start converting your HTML files to PowerPoint presentations.