

Combined Text Directives

Example 1 : Documentation Style

This example demonstrates how to combine different text directives effectively. Use TEXT for normal explanatory prose with proper typography.

Use PURETEXT when showing literal syntax or code examples:

```
hatch run convert examples/hello_world.txt
```

And use LATEX when you need specific LaTeX formatting not available otherwise.

Example 2 : Tutorial Format

Let's explore the natural numbers. In Z notation, we write N for the set of natural numbers: 0, 1, 2, 3, ...

$$\begin{array}{|c} \hline first_five : \mathbb{P} \mathbb{N} \\ \hline \end{array} \\ first_five = \{0, 1, 2, 3, 4\}$$

The above defines a set containing the first five natural numbers. Note how TEXT blocks provide context before and after formal definitions.

Example 3 : Showing Syntax vs Using Syntax

To demonstrate the difference between showing syntax and using it, consider quantifiers.

When we want to use a quantifier in our specification:

$$\forall x : \mathbb{N} \bullet x \geq 0$$

But when we want to show the syntax literally for teaching purposes:

forall x : T — predicate

The PURETEXT block shows the syntax pattern without actually creating a formal quantified expression.

Example 4 : Code Examples with Explanations

The txt2tex tool supports several commands. Here are the most common:

```
# Convert to LaTeX only
```

```
hatch run cli examples/file.txt
```

```
# Convert to PDF (full pipeline)
```

```
hatch run convert examples/file.txt
```

The first command generates only the LaTeX file, while the second runs the complete pipeline including PDF generation.

Tip: Always use the convert command for final output.

Example 5 : Mathematical Explanation

Consider the function $f(n) = n^2$. We can define this formally in Z notation:

$$\begin{array}{c} \textit{square} : \mathbb{N} \longrightarrow \mathbb{N} \\ \hline \forall n : \mathbb{N} \bullet \textit{square}(n) = n * n \end{array}$$

This states that square is a total function from natural numbers to natural numbers.

n	$f(n)$
0	0
1	1
2	4
3	9
4	16

The table above shows the first few values of the square function.

Example 6 : Syntax Reference

Here's a quick reference for text directives:

TEXT: Normal prose with smart quotes and inline math

PURETEXT: Verbatim text without processing

LATEX: Raw LaTeX commands

PAGEBREAK: Force a page break

Example usage:

TEXT: This is a paragraph.

PURETEXT: This is verbatim.

LATEX: \textbf{Bold text}

PAGEBREAK

Example 7 : Best Practices

When writing specifications, follow these guidelines:

1. Use TEXT for all explanatory prose—it provides the best typography
2. Use PURETEXT sparingly, only when you need to show literal syntax or preserve special characters
3. Use LATEX as an escape hatch for formatting not supported by txt2tex
4. Use PAGEBREAK only when you need explicit control over page breaks

By following these practices, your documents will be both readable and maintainable.

Example 8 : Real - World Document Structure

A typical specification document might look like this:

Section 1 : Introduction

Prose introduction explaining the problem domain.

Section 2 : Data Types

[User, Document, Permission]

The system has three basic *types* : *users*, documents, and permissions.

Section 3 : State Schema

```
FileSystem _____  
docs : User →→ P Document  
perms : Document →→ P Permission  
all_docs : P Document  
  
all_docs = ∪(ran docs)  
all_docs ⊆ dom perms
```

The state schema defines the relationships between users, documents, and permissions.

PAGEBREAK

Section 4 : Operations

Now we define the operations on the file system...

This structure—sections with TEXT introductions, formal definitions, and strategic page breaks—creates clear, professional specifications.