

Composition Pipelines

Example 1 : Simple Three - Stage Pipeline

A three-stage numeric processing pipeline.

$$\begin{array}{l} \text{addOne} : \mathbb{N} \rightarrow \mathbb{N} \\ \text{doubleCP} : \mathbb{N} \rightarrow \mathbb{N} \\ \text{squareCP} : \mathbb{N} \rightarrow \mathbb{N} \\ \text{pipeline} : \mathbb{N} \rightarrow \mathbb{N} \\ \hline \forall n : \mathbb{N} \bullet \text{addOne}(n) = n + 1 \\ \forall n : \mathbb{N} \bullet \text{doubleCP}(n) = 2 * n \\ \forall n : \mathbb{N} \bullet \text{squareCP}(n) = n * n \\ \text{pipeline} = \text{squareCP} \circ \text{doubleCP} \circ \text{addOne} \end{array}$$

$$\text{pipeline}(3) = \text{squareCP}(\text{doubleCP}(\text{addOne}(3))) = \text{squareCP}(\text{doubleCP}(4)) = \text{squareCP}(8) = 64$$

Example 2 : String Pipeline

[CharCP]

$$\begin{array}{l} \text{str1} : \text{seq CharCP} \rightarrow \text{seq CharCP} \\ \text{str2} : \text{seq CharCP} \rightarrow \text{seq CharCP} \\ \text{str3} : \text{seq CharCP} \rightarrow \text{seq CharCP} \\ \text{stringPipeline} : \text{seq CharCP} \rightarrow \text{seq CharCP} \\ \hline \text{stringPipeline} = \text{str3} \circ \text{str2} \circ \text{str1} \end{array}$$

Demonstrates composition of string-processing functions.

Example 3 : Best Practices

When building pipelines:

1. Break complex operations into simple stages
2. Each stage should have a single responsibility
3. Test individual functions before composing
4. Use meaningful names for pipeline stages
5. Document the flow direction (comp applies right-to-left)