taskize

Typst Package

Horizontal Columned Lists

A compact list layout for exercises and multiple-choice questions
Version 0.2.6
Nathan Scheinmann

Contents

1. Introduction

`taskize` is a Typst package for horizontal, columned task lists. Items flow left-to-right across columns, making it ideal for exercises, quizzes, and compact lists that need a tidy grid.

1.1. Features

- Indentation-independent parsing
- Horizontal or vertical flow directions
- Flexible column counts and column spanning
- Multiple label formats and custom label functions
- Resume numbering across task blocks
- Fine-grained spacing and alignment controls

1.2. Installation

Import the package in your Typst document:

```
#import "@local/taskize:0.2.5": tasks
```

1.3. Quick Start

Code

```
#tasks[
  + First item
  + Second item
  + Third item
  + Fourth item
]
```

Preview

a) First item   b) Second item
c) Third item   d) Fourth item

2. Basic Usage

2.1. Column Counts

Use `columns:` or shorthand functions (`tasks2`, `tasks3`, `tasks4`) for quick layouts.

Code

```
#tasks(columns: 3)[
  + Option A
  + Option B
  + Option C
  + Option D
  + Option E
  + Option F
]
```

Preview

a) Option A   b) Option B   c) Option C
d) Option D   e) Option E   f) Option F

Code

```
#tasks3[
  + Item 1
  + Item 2
  + Item 3
  + Item 4
  + Item 5
  + Item 6
]
```

Preview

a) Item 1   b) Item 2   c) Item 3
d) Item 4   e) Item 5   f) Item 6

2.2. Labels and Formats

Code

```
#tasks(label: "A)")[+ Alpha + Beta + Gamma]
#tasks(label: "(1)")[+ One + Two + Three]
#tasks(label: n => "Q" + str(n) + ":")[
  + What is 2 + 2?
  + What is the capital of France?
  + What color is the sky?
]
```

Preview

A) Alpha   +   Beta   +   B)
    Gamma
C)
(1) One + Two + Three   (2)
(3)
Q1: What is 2 + 2?   Q2:
Q3: What is the capital of   Q4: What color is the sky?
    France?

2.3. Column Spanning

Use `(N)` to span N columns or `()` to span all columns.

Code

```
#tasks(columns: 3)[
  + (2) Long statement spanning two columns
  + Short
  + () Full width note across all columns
  + Another
]
```

Preview

a) Long statement spanning two   b) Short
   columns
c) Full width note across all columns
d) Another

2.4. Flow Direction

Horizontal is default. Use `flow: "vertical"` to fill columns top-to-bottom.

Code

```
#tasks(columns: 2, flow: "vertical")[
  + a
  + b
  + c
  + d
  + e
  + f
]
```

Preview

a) a   d) d
b) b   e) e
c) c   f) f

3. Configuration

3.1. Global Defaults

Use `tasks-setup()` to set defaults for the rest of the document.

Code

Preview

```
#tasks-setup(
  columns: 3,
  label-format: "1)",
  column-gutter: 1.2em,
  row-gutter: 0.7em,
  label-weight: "bold",
)

#tasks[
  + Item 1
  + Item 2
  + Item 3
  + Item 4
  + Item 5
  + Item 6
]
```

1) Item 1   **2)** Item 2   **3)** Item 3
4) Item 4   **5)** Item 5   **6)** Item 6

3.2. Resume Numbering

Code

Preview

```
#tasks[
  + First
  + Second
  + Third
]

#tasks(resume: true)[
  + Fourth
  + Fifth
]

#tasks-reset()
#tasks[
  + Back to one
  + Two again
]
```

1) First   **2)** Second   **3)** Third
4) Fourth   **5)** Fifth
1) Back to one   **2)** Two again

4. Parameter Reference

The `tasks()` function supports the following parameters:

Parameter	Type	Default	Description
columns	int	2	Number of columns
label	string/function	"a)"	Label format shorthand
label-format	string/function	"a)"	Explicit label format
start	int	1	Starting number
resume	bool	false	Continue numbering from previous tasks
column-gutter	length	1em	Space between columns
row-gutter	length	0.6em	Space between rows
label-width	auto/length	auto	Reserved width for labels
label-align	alignment	right	Label alignment
label-baseline	string/length	"center"	Vertical alignment of labels

label-weight	string	"regular"	Font weight for labels
indent-after-label	length	0.4em	Space after label
indent	length	0pt	Left indentation of the block
above	length	0.5em	Space before the block
below	length	0.5em	Space after the block
flow	string	"horizontal"	"horizontal" or "vertical" flow