sprintf package

`sprintf` package provides `%`-style string formatting for Typst.

It follows Python's old-style formatting semantics and is useful when you want compact format strings for numbers, alignment, and mixed text.

Functions

`sprintf(format: str, ..args) -> str`

- `format`: formatting string that contains `%` directives.
- `..args`: positional values consumed by directives in order.

If wanting named arguments, there are `sprintf-map` function

`sprintf-map(format: str, arg: dictionary) -> str`

`sprintf-map` accepts a format string and a dictionary (mapping) of names to values. Use it when your format string uses named placeholders and you prefer passing a single table of arguments rather than many positional arguments.

Example:

```
#import "@preview/sprintf:0.1.0": *

#let arg = (name: "Eve", score: 9.5)
#sprintf-map("name=%(name)s, score=%(score).1f", arg)
```

Result: `name=Eve, score=9.5`

Quick Start

```
#import "@preview/sprintf:0.1.0": *

#sprintf("Hello, %s!", "Typst")
```

Result: `Hello, Typst!`

Supported Common Directives

The package supports standard printf-style directives commonly used in Python `%` formatting.

- `%s`: string conversion
- `%d`: integer (decimal)
- `%f`: floating-point
- `%x` / `%X`: hexadecimal integer
- `%%`: literal percent sign
- ... (for more specifications, ref https://docs.python.org/3/library/stdtypes.html#printf-style-string-formatting)

Width and precision modifiers are also supported, for example `%8.2f`.

Examples

Strings and Numbers

```
#import "@preview/sprintf:0.1.0": *

#let name = "Ada"
#let score = 98.456

#sprintf("name=%s, score=%.1f", name, score)
```

Result: `name=Ada, score=98.5`

Width and Alignment

```
#import "@preview/sprintf:0.1.0": *

- #sprintf("|%10s|", "cat")
- #sprintf("|%-10s|", "cat")
- #sprintf("|%06d|", 42)
```

Output:

- `|       cat|`
- `|cat       |`
- `|000042|`

Percent Sign

```
#import "@preview/sprintf:0.1.0": *

#sprintf("progress: %d%%", 65)
```

Result: `progress: 65%`

Attribution

This package is based on https://pyformats.nimpylib.org/pyformats/percent.

That library is an implementation in Nim of Python's printf-style string formatting specification: https://docs.python.org/3/library/stdtypes.html#printf-style-string-formatting