

# Shopify Liquid Cheat Sheet

The filters, tags and objects you actually use. Built by Kaspian Fuad, Shopify developer.

## Output & prices

```
{{ product.title }}  
Print any object property.
```

```
{{ product.price | money }}  
Format price in store currency. Prices are in cents.
```

```
{{ product.price | money_with_currency }}  
Adds the ISO code: 10.00 USD.
```

```
{{ 1000 | money }}  
Renders 10.00. money expects cents, not dollars.
```

```
{{ p.compare_at_price | money }}  
Was-price for a sale badge.
```

```
{{ title | default: 'Untitled' }}  
Fallback when the value is empty or nil.
```

## Variables & logic

```
{% assign x = 'blue' %}  
Create or overwrite a variable.
```

```
{% capture block %}...{% endcapture %}  
Assign rendered markup to a variable.
```

```
{% if product.available %}...{% endif %}  
Run a block when a condition is true.
```

```
{% elsif x > y %} / {% else %}  
Extra branches inside an if.
```

```
{% unless product.available %}Sold out{% endunless %}  
if not: cleaner than if == false.
```

```
{% case template.name %}{% when 'index' %}  
Match one value against many.
```

## Loops

```
{% for p in collection.products %}...{% endfor %}  
Iterate an array.
```

```
{% for p in products limit: 4 offset: 2 %}  
Cap and skip items.
```

```
{{ forloop.index }} / .first / .last  
Loop position helpers (1-based).
```

```
{% break %} / {% continue %}  
Stop the loop / skip to the next item.
```

```
{% for i in (1..5) %}  
Loop a numeric range.
```

```
{% for %}...{% else %}Empty{% endfor %}  
else runs when the array is empty.
```

```
{% paginate collection.products by 24 %}  
Chunk large collections. Wrap the loop.
```

## Everyday filters

```
| upcase | downcase | capitalize  
Change case.
```

```
| append: '.html' | prepend: '/'  
Concatenate strings.
```

```
| replace: 'a','b' | remove: 'x'  
String surgery.
```

```
| split: ',' | join: ', '  
String to array and back.
```

```
| size | first | last  
Array and string length / ends.
```

```
| plus: 1 | minus: 1 | times: 2  
Math. Also divided_by, modulo.
```

```
| round: 2 | ceil | floor  
Rounding.
```

```
| json  
Dump an object to inspect it while debugging.
```

## Arrays & collections

```
{{ products | where: 'available' }}  
Keep items whose property is truthy.
```

```
{{ products | where: 'type', 'Shirt' }}  
Keep items matching a value.
```

```
{{ products | map: 'title' }}  
Pull one property from every item.
```

```
{{ array | sort: 'price' }}  
Sort by a property. Also sort_natural.
```

```
{{ array | uniq | reverse }}  
Dedupe / flip order.
```

```
{{ collection.products.size }}  
Count without looping.
```

## Images (never raw CDN URLs)

```
{{ image | image_url: width: 800 }}  
Sized CDN URL. Always set a width.
```

```
{{ image | image_url: width: 800 | image_tag:  
loading: 'lazy', alt: image.alt }}  
Full responsive image tag in one line.
```

```
{{ image | image_url: width: 800, height: 800, crop:  
'center' }}  
Force a square or fixed ratio.
```

```
{{ 'app.css' | asset_url | stylesheet_tag }}  
Link a theme stylesheet.
```

```
{{ 'app.js' | asset_url | script_tag }}  
Link a theme script.
```

## Cart & customer

```
{{ cart.item_count }}  
Items currently in the cart.
```

```
{{ cart.total_price | money }}  
Cart subtotal, formatted.
```

```
{% for item in cart.items %}  
Loop the line items.
```

```
{{ item.quantity }} x {{ item.product.title }}  
Line item output.
```

```
{% if customer %}...{% endif %}  
Is the shopper logged in?
```

```
{{ customer.first_name | default: 'there' }}  
Greet a known customer.
```

## Dates, links, translation

```
{{ 'now' | date: '%b %d, %Y' }}  
Format a date: Jul 03, 2026.
```

```
{{ article.published_at | date: '%B %-d' }}  
No leading zero on the day.
```

```
{{ 'products.general.sale' | t }}  
Output a translated locale string.
```

```
{{ product.title | handleize }}  
Make a URL-safe handle.
```

```
{{ product.url | prepend: shop.url }}  
Build an absolute URL.
```

```
{{ 'Home' | link_to: routes.root_url }}  
Render an anchor tag.
```

## Snippets & sections

```
{% render 'card' %}  
Include a snippet in an isolated scope.
```

```
{% render 'card', product: product %}  
Pass variables in explicitly.
```

```
{% render 'price' for products as p %}  
Loop a snippet over an array.
```

```
{% section 'header' %}  
Render a section by file name.
```

```
{% sections 'footer-group' %}  
Render a section group.
```

```
# include is deprecated. Use render.  
render has isolated scope: fewer bugs.
```

## Objects, metafields, settings

```
{{ product.metafields.custom.subtitle.value }}  
Read a metafield. .value unwraps typed ones.
```

```
{{ shop.name }} / shop.url / shop.currency  
Store globals.
```

```
{{ section.settings.heading }}  
A section's schema setting.
```

```
{{ block.settings.text }}  
A block's schema setting.
```

```
{{ request.page_type }} / template.name  
What page am I on?
```

```
{{ canonical_url }} / {{ routes.root_url }}  
SEO + link building blocks.
```

## Whitespace, comments, speed

```
{%- assign x = 1 -%}  
Hyphens strip surrounding whitespace.
```

```
{{- x -}}
```

*Same for output tags.*

```
{% comment %}...{% endcomment %}  
Block comment.
```

```
{% # a quick note %}  
Inline comment (newer Liquid).
```

```
{% raw %}{{ literal }}{% endraw %}  
Output Liquid syntax as text.
```

Avoid nested product loops.  
*Use where / map / paginate to cut server time.*