



Templating: nunjucks

Lesson 8



Lesson Plan

1

What is templates in HTML and why we need it

2

Nunjucks template language

3

Templating features:

- set
- for
- if
- include
- extends

Templating

is a way to automate HTML

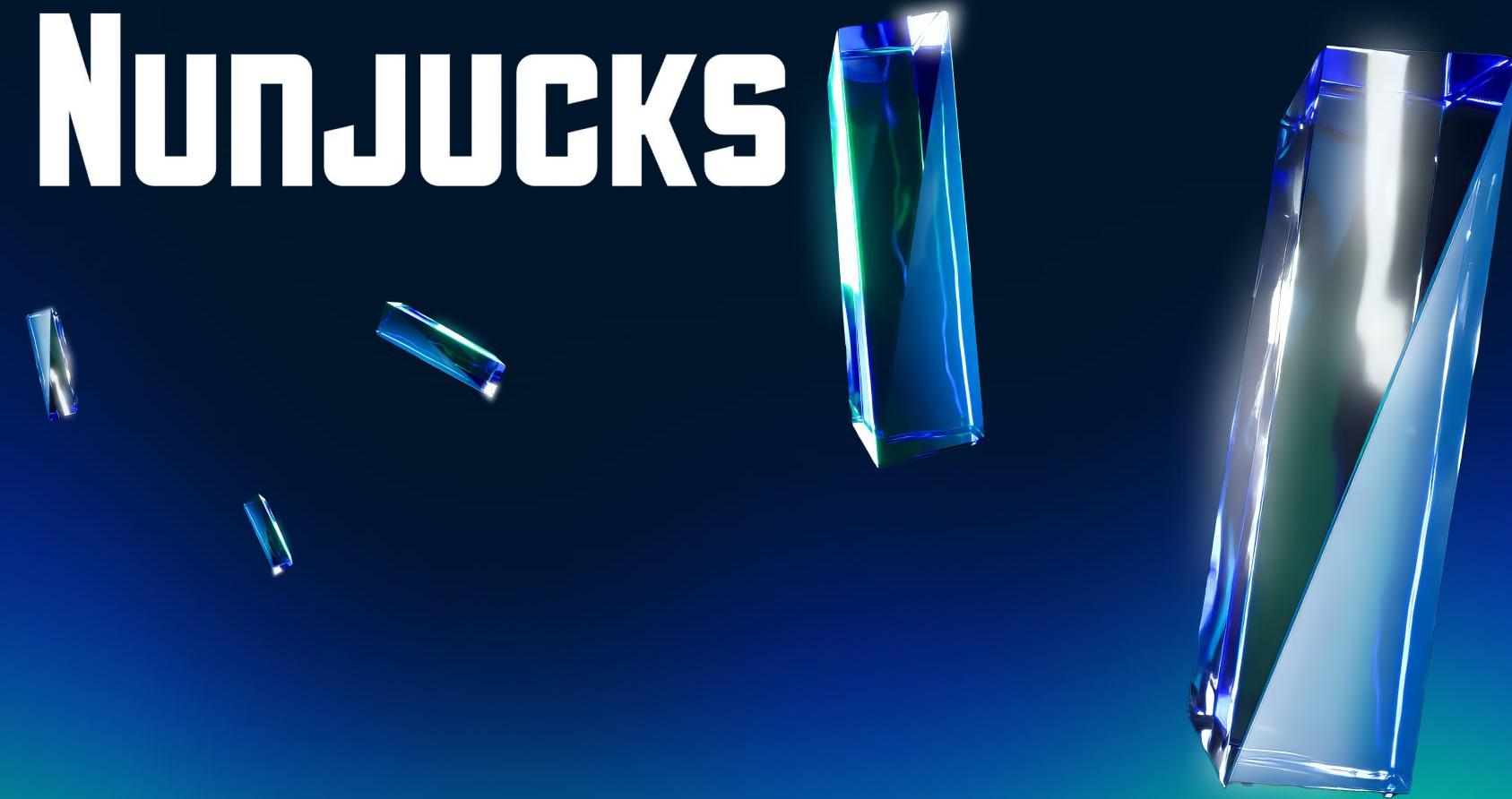
- helps to avoid code duplication
- simplifies code maintenance
- supports providing dynamic data

```
{% extends "base.html" %}

{% block header %}
<h1>{{ title }}</h1>
{% endblock %}

{% block content %}
<ul>
    {% for name, item in items %}
        <li>{{ name }}: {{ item }}</li>
    {% endfor %}
</ul>
{% endblock %}
```

NUNJUCKS



Nunjucks is a templating language

- Reuse parts of UI as components
- Reuse and easily control a content with variables
- Repeat repetitive things with cycles
- Show different things depending on data

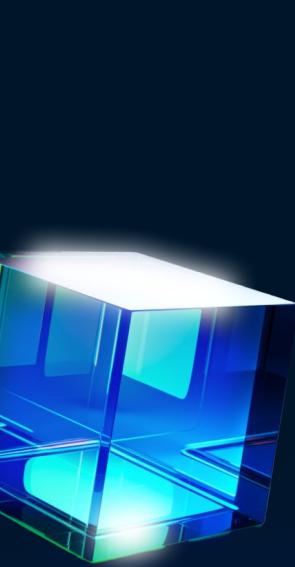
Nunjucks in file system

```
└── src/
    └── templates/      # Folder with all templates
        ├── pages/       # Pages (index/index.njk, blog/blog.njk, etc)
        ├── layouts/      # Layouts (base.njk)
        └── partials/      # Repeating parts (header.njk, footer.njk, card.njk)
```

Nunjucks

```
└── src/
    ├── templates/          # Folder with all templates
    │   ├── pages/           # Pages (index.njk, about.njk, etc)
    │   ├── layouts/          # Layouts (base.njk)
    │   └── partials/         # Repeating parts (header.njk, footer.njk, card.njk)
    ├── styles/              # Styles (CSS/SCSS)
    │   ├── pages/            # Entry points (styles.scss)
    │   └── components/       # Files for BEM blocks
    ├── images/               # Images
    │   ├── static/            # convert to webp and avif, compress
    │   └── sprite/            # combine into SVG file sprite.svg
    └── gulpfile.js           # Gulp settings
└── public/                # Resulting build folder
```

Templating features



Nunjucks

Variables



{% set %} a variable

and then you'll have an ability to use it somewhere else

index.njk

```
{% set title = "Test page" %}
```



{{ variables }} are containers for data

You may use it to substitute a predefined data
on the place where you used a variable

```
index.njk
{% set title = "Test page" %}
{{ title }}
```

2

Content from `data.json`

```
src/
  └── templates/
    ├── layouts/
    ├── partials/
    └── pages/
      └── index/
        ├── index.njk
        └── data.json
```

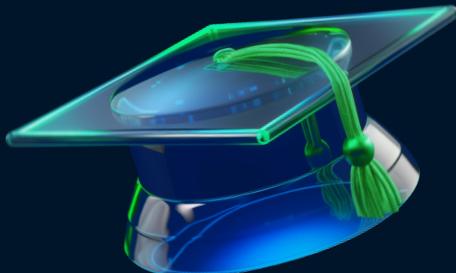
`data.json`

```
{  
  "title": "Binabox",  
  "path" : "/",  
  "key"  : "value"  
}
```

1. File should be started with open curly bracket
2. and ended with closing curly bracket
3. Each **object** key and value should be in quotes*
4. Keys and values should be separated by colon

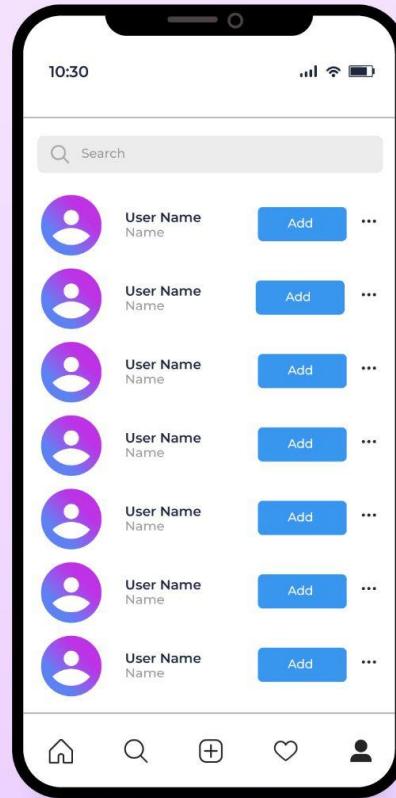
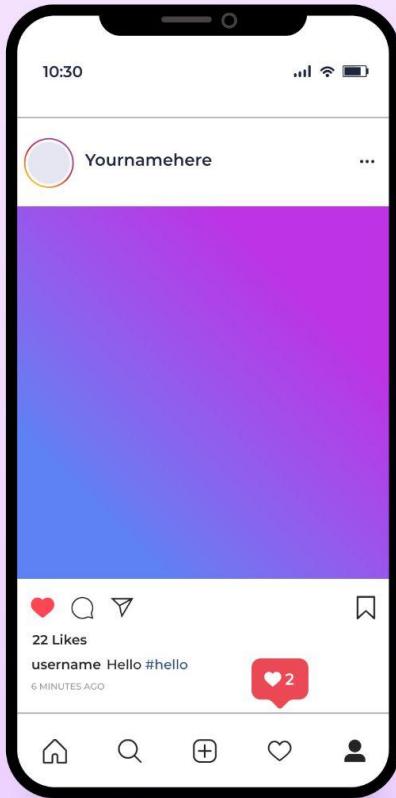
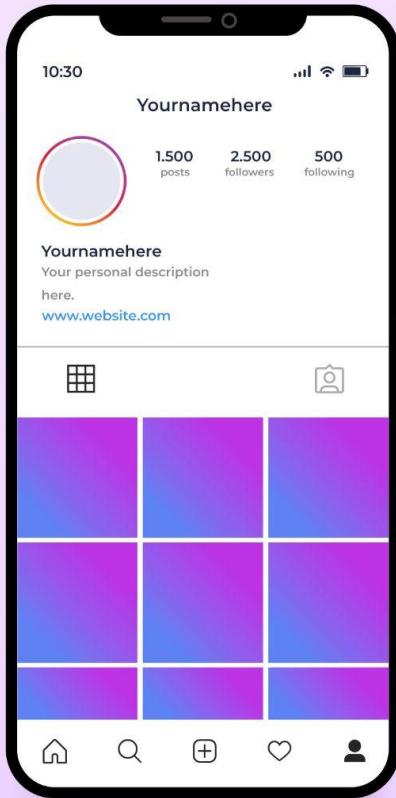
When you need to use variables

- when you're getting data from different sources
- when data may be changed depending on any conditions and should be replaced automatically in different places
- or always to separate data from views and add make markup logic more transparent





Nunjucks
Cycles



{% for %} cycle

will allow you to repeat HTML block
with different arrays of data

```
/src/templates/pages/index/index.njk
{% for icon in networks %}
    
{% endfor %}
```



```
/public/index.html



```

data.json

```
{
    "networks": [
        "facebook",
        "telegram",
        "instagram",
    ]
}
```

Our team



Jason Smith
Co - Founder

[!\[\]\(f9640f02d06b604c50c1e3a558a31388_img.jpg\)](#) [!\[\]\(d9d98187a25260cd1316acead95958b1_img.jpg\)](#) [!\[\]\(5e91ce58ee8e9ab2222ea2dcf7128599_img.jpg\)](#) [!\[\]\(0fee90055c33b690097ce6e49c052bde_img.jpg\)](#) [!\[\]\(0b90a7fd1c0b66996870b89b061dc7bd_img.jpg\)](#)



Jason Smith
Co - Founder

[!\[\]\(16aeb3237f754b7599789ef654002ff2_img.jpg\)](#) [!\[\]\(4bedab4577924086e5ae6352aa0c2ad0_img.jpg\)](#) [!\[\]\(5462a62144ec6ad15b20b37b7ca1cfd9_img.jpg\)](#) [!\[\]\(d31fb64dbb514a38bb02cd3a863f2d7f_img.jpg\)](#) [!\[\]\(76c209a6b463cb2a110b76c84ef51303_img.jpg\)](#)



Jason Smith
Co - Founder

[!\[\]\(0ebf3f9f6e49e9383215c359d6d92943_img.jpg\)](#) [!\[\]\(7e13409480a9e0a00ad84d3a2a18a0f6_img.jpg\)](#) [!\[\]\(c3c7aa4d731fe2b03345c11e9fdc1354_img.jpg\)](#) [!\[\]\(6a5b34b8bdb086bb967bfa9ad5d6fba8_img.jpg\)](#) [!\[\]\(b6bc8d2996cef55079081966d1eac75c_img.jpg\)](#)



Jason Smith
Co - Founder

[!\[\]\(8832aca84a80d65d1b480375fc2819bd_img.jpg\)](#) [!\[\]\(8969bf5f67df0b1b5258b35f1250a7b5_img.jpg\)](#) [!\[\]\(a2479761f0b82d27d551c6fc1a610de1_img.jpg\)](#) [!\[\]\(0e2a28f24ba2f95bf5a629add518298d_img.jpg\)](#) [!\[\]\(3bc36ed0c15e90cb836d4ca7521676c1_img.jpg\)](#)



Jason Smith
Co - Founder

[!\[\]\(e79b94ae013d5475cd2f9cbd052d0d86_img.jpg\)](#) [!\[\]\(28aff82a7da938aa77773aedc99d17c0_img.jpg\)](#) [!\[\]\(d178f8f58e83d93033855a02b799f195_img.jpg\)](#) [!\[\]\(d49c07562f397d2c322fcb621fe438aa_img.jpg\)](#) [!\[\]\(06ee554930e261006749deb68a3ce44c_img.jpg\)](#)



Jason Smith
Co - Founder

[!\[\]\(397b56ec192bbefdefc1bdb72ce44c85_img.jpg\)](#) [!\[\]\(7b2e82d8072b00572d44603f9d237bea_img.jpg\)](#) [!\[\]\(ccbf429d3e7f4fd7c131d5528413337d_img.jpg\)](#) [!\[\]\(869c3afa27c90a5e1b28452d6584e735_img.jpg\)](#) [!\[\]\(df527a30d78b02cef87f0d02e4d1b191_img.jpg\)](#)



Jason Smith
Co - Founder

[!\[\]\(92f3141e8992f2c6864cc606eec34772_img.jpg\)](#) [!\[\]\(9e10310185461fcc91ff02ba1ee6d77b_img.jpg\)](#) [!\[\]\(75289230efaa189e88ddfbd130886b9f_img.jpg\)](#) [!\[\]\(f27139b3ff08ee92df7d67089c58d094_img.jpg\)](#) [!\[\]\(0edecd18c4870df9eb72723710119fb0_img.jpg\)](#)

{% for %} with nested objects

```
/src/templates/pages/index/index.njk
{% for person in team %}
- {{ person.name }}
- {{ person.role }}
- {{ person.avatar }}
{% endfor %}
```



```
/public/index.html
- Jason Smith
- Co-Founder
- jason-smith.jpg
- Anna Klaus
- Engineer
- anna-klaus.jpg
```

data.json

```
{
  "title": "Binabox",
  "path" : "/",
  "team" : [
    {
      "name": "Jason Smith",
      "role": "Co-Founder",
      "avatar":
        "jason-smith.jpg",
    },
    {
      "name": "Anna Klaus",
      "role": "Engineer",
      "avatar":
        "anna-klaus.jpg",
    }
  ]
}
```

/src/templates/partials/nav.njk

Home

Blog

Catalog

{% for %} with nested objects

/src/templates/pages/blog/blog.njk

```
<ul>
  {% for item in nav %}
    <li><a href="{{ item.url }}">{{ item.text }}</a></li>
  {% endfor %}
</ul>
```



/public/blog.html

```
<ul>
  <li><a href="/">Home</a></li>
  <li><a href="/blog/">Blog</a></li>
  <li><a href="/catalog/">Catalog</a></li>
</ul>
```

/src/templates/pages/blog/data.json

```
{
  "title": "Blog",
  "path": "/blog/",
  "nav": [
    {
      "url": "/",
      "text": "Home"
    },
    {
      "url": "/blog/",
      "text": "Blog"
    },
    {
      "url": "/catalog/",
      "text": "Catalog"
    }
  ]
}
```

{% for %} with objects unpacking

/src/templates/pages/blog/blog.njk

```
<ul>
  {% for url, text in nav %}
    <li><a href="{{ url }}>{{ text }}</a></li>
  {% endfor %}
</ul>
```



/public/blog.html

```
<ul>
  <li><a href="/">Home</a></li>
  <li><a href="/blog/">Blog</a></li>
  <li><a href="/catalog/">Catalog</a></li>
</ul>
```

/src/templates/pages/blog/data.json

```
{
  "title": "Blog",
  "path": "/blog/",
  "nav": [
    {
      "url": "/",
      "text": "Home"
    },
    {
      "url": "/blog/",
      "text": "Blog"
    },
    {
      "url": "/catalog/",
      "text": "Catalog"
    }
  ]
}
```

Home

Blog

Catalog

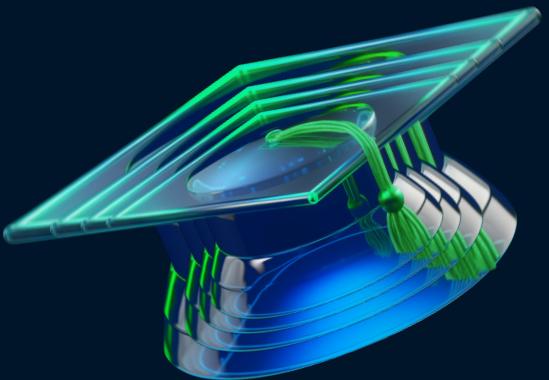


When you need to use cycles

- when you have repetitive content with typical design
- when items amount is known
- to not write annoying repetitive code

Examples:

- navigation items
- cards
- social icons
- accordion items
- categories list
- article tags
- catalog filter list
- etc



The image shows a 4x4 grid of cards, each containing a small image, a price (\$199), a product name ('Name Product'), and a small red badge indicating status ('New', 'Sold out', or 'New'). The cards are organized into four distinct sections: products, categories, profiles, and another set of profiles.

Category	Content	Image	Price	Status
Product	Skeleton		\$199	New
Product	Black Cat		\$199	New
Product	Scissors		\$199	Sold out
Product	Book		\$199	New
Category	High quality		—	—
Category	Big collection		—	—
Category	Top Resource		—	—
Category	Big community		—	—
Profile	Jason Smith, Co-Founder		—	—
Profile	Jason Smith, Co-Founder		—	—
Profile	Jason Smith, Co-Founder		—	—
Profile	Jason Smith, Co-Founder		—	—
Profile	Jason Smith, Co-Founder		—	—
Profile	Jason Smith, Co-Founder		—	—
Profile	Jason Smith, Co-Founder		—	—
Profile	Jason Smith, Co-Founder		—	—

Nunjucks
Conditions



{% if %} condition {% endif %}

check if variable is equal to something another

```
{% if something %}  
    something  
{% endif %}"
```

```
{% if something == another %}  
    something  
{% endif %}"
```

{% if %} condition {% endif %}

check if variable is equal to something

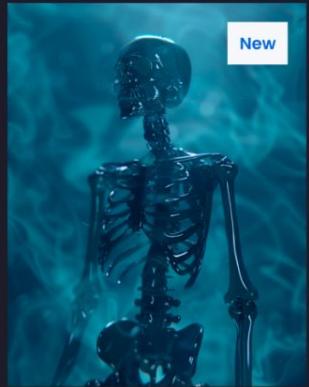
```
/src/templates/pages/blog/blog.njk
{% for item in nav %}
<a href="{{ item.url }}"
class="nav__link {{ if nav.url == url
}}nav__link--active{{ endif }}>
{{ item.text }}</a>
{% endfor %}
```



```
/public/blog.html
<li><a class="nav__link"
href="/">Home</a></li>
<li><a href="/blog/" class="nav__link
nav__link--active">Blog</a></li>
<li><a class="nav__link"
href="/catalog/">Catalog</a></li>
```

/src/templates/pages/blog/data.json

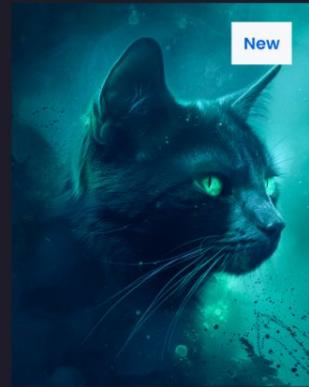
```
{
  "title": "Blog",
  "url": "/blog/",
  "nav": [
    {
      "url": "/",
      "text": "Home"
    },
    {
      "url": "/blog/",
      "text": "Blog"
    },
    {
      "url": "/catalog/",
      "text": "Catalog"
    }
  ]
}
```



\$199

Name Product

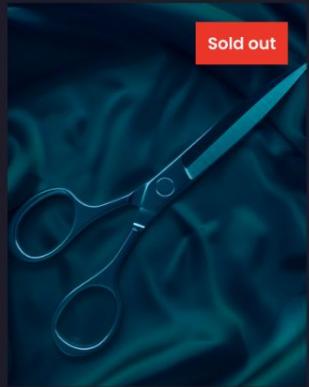
New



\$199

Name Product

New



\$199

Name Product

Sold out



\$199

Name Product

New



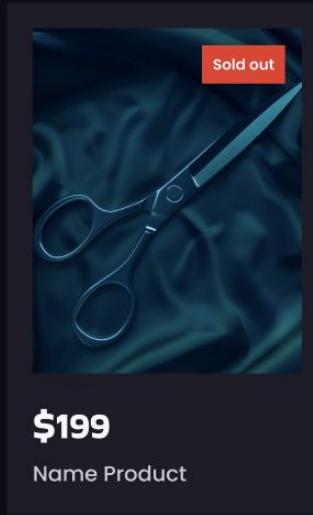
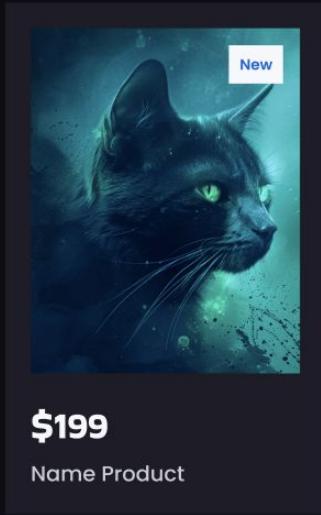
\$199

Name Product

New

/src/templates/pages/blog/data.json

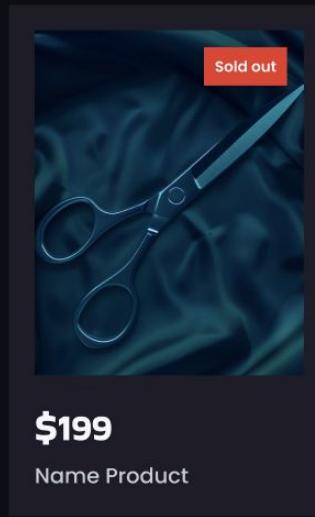
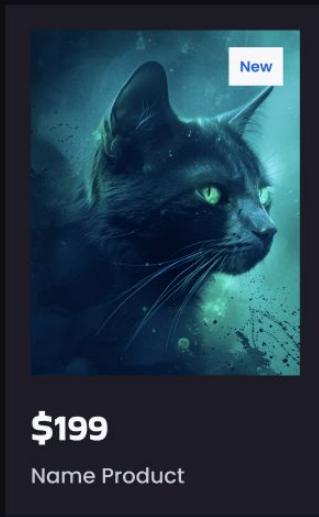
```
"catalog-items": [
    {
        "label": "New",
        "name": "Name Product",
        "price": "199",
        "image": "cat.jpg",
    },
    {
        "label": "Sold out",
        "name": "Name Product",
        "price": "199",
        "image": "cat.jpg",
    }
]
```



```
<p class="card__label
  {% if catalog-item.label == "Sold out" %}
      card__label--active
  {% endif %}"
>{{ catalog-item.label }}</p>
```

/src/templates/pages/blog/data.json

```
"items": [
  {
    "label": "New",
    "state": "new",
    "price": "199",
    "name": "Name Product",
    "image": "cat.jpg",
  },
  {
    "label": "Sold out",
    "state": "sold-out",
    "price": "199",
    "name": "Name Product",
    "image": "cat.jpg",
  }
]
```



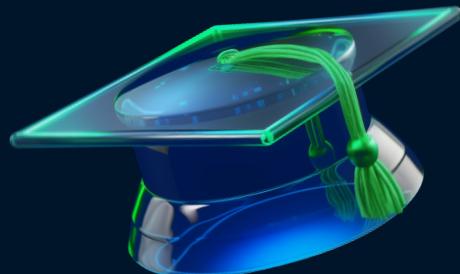
```
<p class="card__label card__label--{{ item.state }}>
{{ item.label }}</p>
```

When you need to use condition

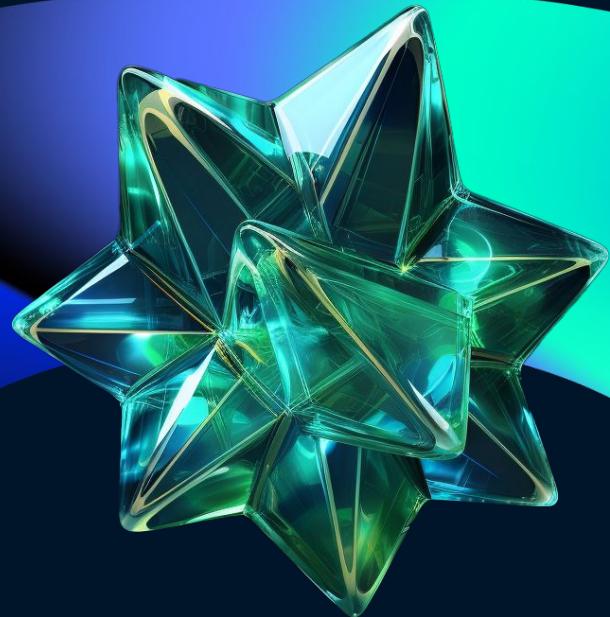
- when something should be changed in different cases

Examples:

- Highlight current page in the navigation menu
- Choose current radio button in filters list
- Show different tag on a card



BREAK TIME



Nunjucks
Includes



{% include %}

allows to put a template on desired place

```
/src/templates/layout/base.njk
{% for item in nav %}
<a href="{{ item.url }}" class="nav_link {% if nav.url == url
%}nav_link--active{% endif %}">
{{ item.text }}</a>
{% endfor %}
```



```
/public/blog.html
<li><a href="/">Home</a></li>
<li><a href="/blog/" class="active">Blog</a></li>
<li><a href="/catalog/">Catalog</a></li>
```

```
/src/templates/partials/header.njk
```



Home Blog Catalog Contact Us

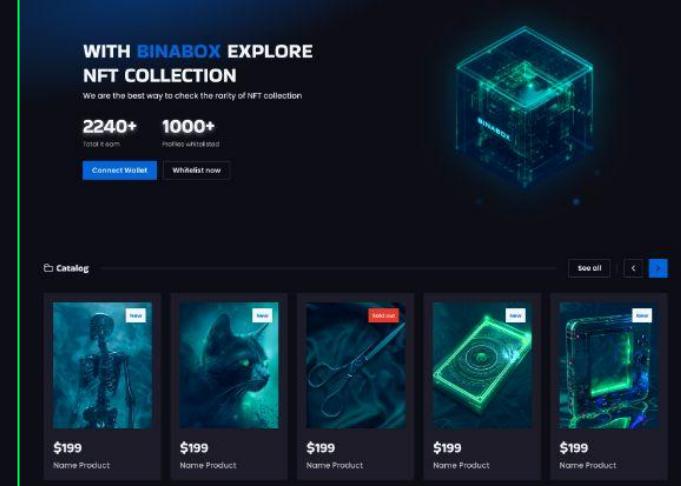
Search

Discord

Connect

```
{% include "/src/templates/partials/header.njk" %}
```

```
/src/templates/layouts/base.njk
```



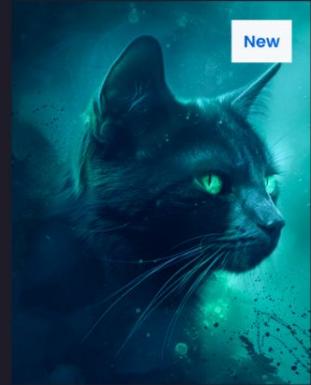
```
{% for item in nav %}  
{% include "/src/templates/partials/card.njk" %}  
{% endfor %}
```



\$199

Name Product

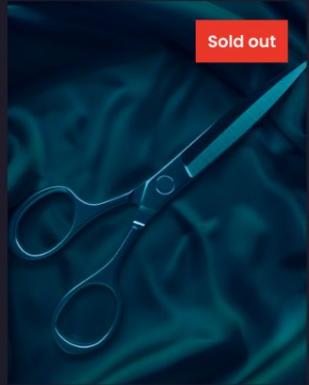
New



\$199

Name Product

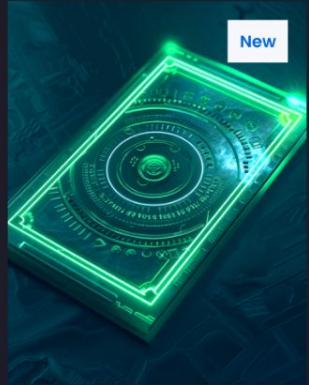
New



Sold out

\$199

Name Product



New

\$199

Name Product

New



\$199

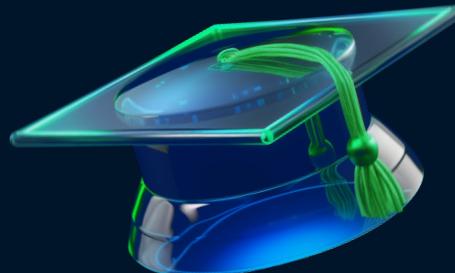
Name Product

When you may want to use includes

- to reduce complexity of a component
- to name a component like a bem block

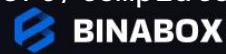
Examples:

- header
- navigation
- card
- section





Nunjucks
Extends

[Home](#)[Blog](#)[Catalog](#)[Contact Us](#) Search [Discord](#) [Connect](#)

```
{% block "content" %}<h1>base</h1>{% endblock %}
```



We are the best way to check the rarity of NFT collection.

Subscribe[Home](#)[Blog](#)[Catalog](#)[Contact Us](#)**Company**[Terms Conditions](#)[Privacy Policy](#)[Whitepaper](#)[Help Center](#)

{% extends %} and {% block %}

replacing blocks by names in a template

```
/src/templates/layouts/base.njk
<body>
  {% block "content" %}<h1>base</h1>{% endblock %}
</body>
```

```
/src/templates/pages/index/index.njk
{% extends "layouts/base.njk" %}
  {% block "content" %}<h1>index</h1>{% endblock %}
```



```
/public/index.html
<body>
  <h1>index</h1>
</body>
```

```
/src/templates/pages/blog/blog.njk
{% extends "layouts/base.njk" %}
any text
```



```
/public/blog.html
<body>
  <h1>base</h1>
</body>
```

Pass page data through layout into includes

```
/src/templates/layouts/base.njk
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <title>{{ title or "Binabox" }}</title>
</head>
<body>
  {% include "partials/header.njk" %}
  {% block "content" %}<h1>base</h1>{% endblock %}
  {% include "partials/footer.njk" %}
</body>
</html>

/src/templates/pages/blog/blog.njk
{% extends "layouts/base.njk" %}
{% block "content" %}<h1>blog</h1>{% endblock %}
```



/src/templates/pages/blog/data.json

```
{ "title": "Blog", "path": "/" }
```

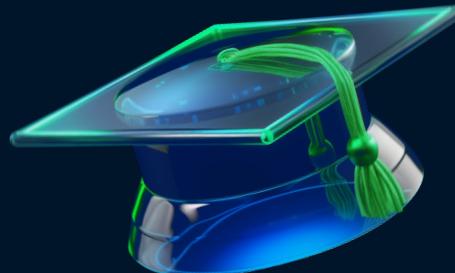
```
/public/blog.html
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <title>Blog</title>
</head>
<body>
  {% include "partials/header.njk" %}
  {% block "content" %}<h1>blog</h1>{% endblock %}
  {% include "partials/footer.njk" %}
</body>
</html>
```

When you may want to use extends

when you need to have some places that should be used in other parts of your code

Examples:

- layouts
- cards





Summarize

Let's check figma

The screenshot shows the BINABOX homepage with a dark blue background. At the top, there is a navigation bar with the BINABOX logo, a search bar, and social media links for Discord and Connect.

**WITH BINABOX EXPLORE
NFT COLLECTION**

We are the best way to check the rarity of NFT collection

2240+ Total it earn **1000+** Profiles whitelisted

[Connect Wallet](#) [Whitelist now](#)

A large, glowing, translucent blue cube is centered on the page, representing the NFT collection. Below the cube, there is a section titled "Catalog" with five thumbnail images of NFTs. The first thumbnail shows a skeleton, the second shows a black cat, the third shows a pair of scissors, the fourth shows a card, and the fifth shows a glowing cube. Each thumbnail has a "New" or "Sold out" status indicator.

[See all](#) | < >



QUESTIONS?



Quality Criteria for HTML Course

- ❤️ Mandatory for passing the course
- 💛 Required for the highest grade
- 💚 Optional

- ❤️ Gulp is used as a build tool
- ❤️ Public and node_modules folders should not be in the GitHub repository
- ❤️ The project builds correctly
- ❤️ No errors (error) in the console during the build process
- ❤️ The project is structured using Nunjucks templates
- 💛 No warnings (warning) in the console during the build process
- 💛 Loops, variables, and dynamic data are used in Nunjucks

Homework

Part 1: Set Up Gulp Build

**Copy the necessary Gulp configuration files
(gulpfile.js, package.json, etc.) to your project.**

Organize your project files:

- Move all images to the src/images/ folder.**
- Copy the styles from your existing CSS file and place them in src/styles/style.scss.**



Part 2: Create HTML Templates Using Nunjucks

1. Set up the templates folder structure:
 - `src/templates/` → Main folder for all `.njk` files.
 - `src/templates/layouts/` → Base structure for pages (`base.njk`).
 - `src/templates/pages/` → Separate folder for each page (`index.njk`, `about.njk`, etc.).
 - `src/templates/partials/` → Components (e.g., header, footer, product cards).
 - `src/templates/pages/*/data.json` → Store dynamic data (e.g., product lists, user profiles).
2. Implement the Nunjucks templating system:
 - Create a base template (`base.njk`) with `{% block %}` sections for content.
 - Use `{% extends %}` to inherit the layout structure.
 - Include reusable components (header, footer) with `{% include %}`.
 - Work with variables and loops (`{% for %}`, `{% if %}`) to generate dynamic content.

Please fill out the feedback form
It's very important for us



THANK YOU!
Have a good evening!