Learning Twig: Basic Syntax and Features

1. Introduction to Templating

Templating in web development refers to the segregation of HTML and PHP (or any other server-side language) code. It helps in organizing your code, making it more readable and maintainable. Twig is a templating engine for the PHP programming language.

2. Getting Started with Twig

To install Twig, you can use Composer:

```bash
composer require 'twig/twig:^2.0'
```

After installing Twig, you can initialize it in your PHP code as follows:

```php
require 'vendor/autoload.php';
$loader = new \Twig\Loader\FilesystemLoader('templates');
$twig = new \Twig\Environment($loader);
```

3. Variables

Variables in Twig are enclosed within double curly braces. For example, to display a variable named 'name':

```twig
{{ name }}
```

4. Control Structures

Twig provides control structures like 'if', 'else', and 'for'.

Example using 'if' and 'else':

```twig
{% if name == 'John' %}
```

```
  Hello, John!
{% else %}
  Hello, Guest!
{% endif %}
```

5. Filters

Filters in Twig are applied to variables using the pipe ('|') symbol. For example, to convert a variable to uppercase:

```twig
{{ name|upper }}
```

Learning Twig: Intermediate Topics

1. Template Inheritance

Template inheritance allows you to build a base skeleton template that has blocks that child templates can override.

For example, a base template might look like this:

```twig
<!DOCTYPE html>
<html>
<head>
    <title>{% block title %}Default Title{% endblock %}</title>
</head>
<body>
    {% block content %}{% endblock %}
</body>
</html>
```And a child template could override the blocks like this:

```twig
{% extends 'base.html' %}
{% block title %}My Page{% endblock %}
{% block content %}Page content here.{% endblock %}
```

2. Including Templates

The 'include' function allows you to include another Twig file within a template.

Example:

```twig
{% include 'header.html' %}
Body content here.
```

{% include 'footer.html' %}

```

### 3. Macros

Macros are similar to functions in regular programming and allow you to package multiple statements into a reusable chunk.

Example:

```twig

{% macro input(name, value) %}

<input type="text" name="{{ name }}" value="{{ value }}">

{% endmacro %}

```You can then import and use the macro like this:

```twig

{% import _self as forms %}

{{ forms.input('username', 'john.doe') }}

```

4. Debugging

Twig provides a 'dump' function that can help you debug variables.

Example:

```twig

{{ dump(user) }}

```

5. Custom Filters and Functions

You can define your own filters and functions in Twig. To create a custom filter, you'd typically do this in your PHP setup code:

```php

$twig->addFilter(new \Twig\TwigFilter('rot13', function ($string) {

    return str_rot13($string);

}));

```Then you can use this filter in your templates:

```twig

{{ 'hello'|rot13 }}

```
```

Learning Twig: Advanced Topics

1. Namespaces

Namespaces in Twig help to organize templates better and are especially useful in larger projects.

You can define a namespace in your PHP setup code like so:

```php
$loader->addPath('/path/to/templates', 'my_namespace');
```And then use it in your Twig templates like this:

```twig
{% include '@my_namespace/template.twig' %}
```

2. Caching

Caching can improve the performance of your Twig templates. By default, Twig caches compiled templates on the filesystem.

You can set a custom cache directory like so:

```php
$twig = new \Twig\Environment($loader, [
    'cache' => '/path/to/compilation_cache',
]);
```

3. Escaping

Twig has built-in escaping for better security. By default, all variables are HTML-escaped:

```twig
{{ user_content }}  {# automatically HTML-escaped #}
```You can also manually specify the escaping strategy:

```twig
{{ user_content|escape('js') }}  {# JavaScript-escaped #}
```

4. Internationalization

Twig doesn't directly provide internationalization features, but it can be used with tools like Symfony's Translation component to support multiple languages.

Example with Symfony Translation component:

```twig
{{ 'hello'|trans }}
```