Force a naming convention for templates (template-names)

When it comes to naming templates there are multiple naming conventions available. Enforce one of them with this rule.

Rule Details

This rule aims to enforce one naming convention for template names is used consistently. It does this by checking references to the template from the JavaScript code.

It offers three different naming conventions, one of which can be chosen through the rule options.

The following patterns are considered warnings:

```
/*eslint meteor/template-names: [2, "camel-case"]*/
Template.foo_bar.onCreated
Template.foo_bar.onRendered
Template.foo_bar.onDestroyed
Template.foo_bar.events
Template.foo_bar.helpers


Template.foo_bar.onCreated()
/* .. */

Template.FooBar.onCreated
/* .. */
```

The following patterns are not warnings:

```
/*eslint meteor/template-names: [2, "camel-case"]*/
Template.fooBar.onCreated
Template.fooBar.onRendered
Template.fooBar.onDestroyed
Template.fooBar.events
Template.fooBar.helpers

/*eslint meteor/template-names: [2, "pascal-case"]*/
Template.FooBar.onCreated
/* .. */

/*eslint meteor/template-names: [2, "snake-case"]*/
Template.foo.onCreated
Template.foo_bar.onCreated
```

Options

This rule accepts a single options argument with the following defaults:

```
{
    "rules": {
        "template-names": [2, "camel-case"]
    }
}
```

The second argument can have the following values:

- `camel-case`
- `pascal-case`
- `snake-case`

Limitations

This rule can not warn for templates which are never referenced in JavaScript.

When Not To Use It

If you are not using Blaze templates, it is okay to turn this rule off.