

Flask

```
from utils import to_pretty_json
```

```
app = Flask(__name__)
```

```
...
```

```
app.jinja_env.filters['to_pretty_json'] = to_pretty_json
```

Using Flask()

- Filters are part of the Flask application not the Blueprint
- You use the `jinja_env.filters` to create filters
- Follows the dictionary interface



{%blockcontent%}

<odiv>

<h2>Profile</h2>

< / d i v >

<pre>{session['user'|to_pretyjsn]}</pre>

{%extends 'base.html'%}

< / d i v >

<div>

<div class="json-container">

<p>Welcome { {request.session['user_info'].name} }!</p>

<p>Here is your profile information:</p>

}%endblock%

< / d i v >

templates

automatically available for

Using Flask()

• After a filter is created it is



app.injaenv.filters['to_pretty_json'] = to_pretty_json

from util import to_pretty_json

app = Flask(name)

- Filters are part of the Flask

application not the Blueprint

• Follows the dictionary

creatable filters

interfere

jinjanev.filters to

• You use the



app.inj_aenv.filters['to_pretty_json'] == to_pretty_json

from util import to_prety_n

app = Flask(__name__)

