

Flask – Extensions

Flask is often referred to as a micro framework, because a core functionality includes WSGI and routing based on **Werkzeug** and template engine based on **Jinja2**. In addition, Flask framework has support for cookie and sessions as well as web helpers like **JSON**, static files etc. Obviously, this is not enough for the development of a full-fledged web application. This is where the Flask extensions come in picture. Flask extensions give extensibility to Flask framework.

There are a large number of Flask extensions available. A Flask extension is a Python module, which adds specific type of support to the Flask application. Flask Extension Registry is a directory of extensions available. The required extension can be downloaded by **pip** utility.

In this tutorial, we will discuss the following important Flask extensions –

Flask Mail – provides SMTP interface to Flask application

Flask WTF – adds rendering and validation of WForms

Flask SQLAlchemy – adds SQLAlchemy support to Flask application

Flask Sijax – Interface for Sijax - Python/jQuery library that makes AJAX easy to use in web applications

Each type of extension usually provides extensive documentation about its usage. Since an extension is a Python module, it needs to be imported for it to be used. Flask extensions are generally named as flask-foo. To import,

```
from flask_foo import [class, function]
```

For versions of Flask later than 0.7, you can also use the syntax –

```
from flask.ext import foo
```

For this usage, a compatibility module needs to be activated. It can be installed by running flaskext_compat.py

```
import flaskext_compat
flaskext_compat.activate()
from flask.ext import foo
```

Revision #1

Created 14 December 2022 11:26:55 by Admin

Updated 14 December 2022 11:27:11 by Admin