

Patterns for Flask

Certain features and interactions are common enough that you will find them in most web applications. For example, many applications use a relational database and user authentication. They will open a database connection at the beginning of the request and get the information for the logged in user. At the end of the request, the database connection is closed.

These types of patterns may be a bit outside the scope of Flask itself, but Flask makes it easy to implement them. Some common patterns are collected in the following pages.

- [Larger Applications](#)
 - [Simple Packages](#)
 - [Working with Blueprints](#)
- [Application Factories](#)
 - [Basic Factories](#)
 - [Factories & Extensions](#)
 - [Using Applications](#)
 - [Factory Improvements](#)
- [Application Dispatching](#)
 - [Working with this Document](#)
 - [Combining Applications](#)
 - [Dispatch by Subdomain](#)
 - [Dispatch by Path](#)
- [Implementing API Exceptions](#)
 - [Simple Exception Class](#)
 - [Registering an Error Handler](#)
 - [Usage in Views](#)
- [Using URL Processors](#)
 - [Internationalized Application URLs](#)
 - [Internationalized Blueprint URLs](#)
- [Deploying with Setuptools](#)
 - [Basic Setup Script](#)
 - [Tagging Builds](#)
 - [Distributing Resources](#)
 - [Declaring Dependencies](#)
 - [Installing / Developing](#)

- Deploying with Fabric
 - Creating the first Fabfile
 - Running Fabfiles
 - The WSGI File
 - The Configuration File
 - First Deployment
 - Next Steps
- Using SQLite 3 with Flask
 - Connect on Demand
 - Easy Querying
 - Initial Schemas
- SQLAlchemy in Flask
 - Flask-SQLAlchemy Extension
 - Declarative
 - Manual Object Relational Mapping
 - SQL Abstraction Layer
- Uploading Files
 - A Gentle Introduction
 - Improving Uploads
 - Upload Progress Bars
 - An Easier Solution
- Caching
- View Decorators
 - Login Required Decorator
 - Caching Decorator
 - Templating Decorator
 - Endpoint Decorator
- Form Validation with WTForms
 - The Forms
 - In the View
 - Forms in Templates
- Template Inheritance
 - Base Template
 - Child Template
- Message Flashing
 - Simple Flashing

- [Flashing With Categories](#)
- [Filtering Flash Messages](#)
- [AJAX with jQuery](#)
 - [Loading jQuery](#)
 - [Where is My Site?](#)
 - [JSON View Functions](#)
 - [The HTML](#)
- [Custom Error Pages](#)
 - [Common Error Codes](#)
 - [Error Handlers](#)
 - [Returning API errors as JSON](#)
- [Lazily Loading Views](#)
 - [Converting to Centralized URL Map](#)
 - [Loading Late](#)
- [MongoDB with MongoEngine](#)
 - [Configuration](#)
 - [Mapping Documents](#)
 - [Creating Data](#)
 - [Queries](#)
 - [Documentation](#)
- [Adding a favicon](#)
 - [See also](#)
- [Streaming Contents](#)
 - [Basic Usage](#)
 - [Streaming from Templates](#)
 - [Streaming with Context](#)
- [Deferred Request Callbacks](#)
- [Adding HTTP Method Overrides](#)
- [Request Content Checksums](#)
- [Celery Background Tasks](#)
 - [Install](#)
 - [Configure](#)
 - [An example task](#)
 - [Run a worker](#)
- [Subclassing Flask](#)
- [Single-Page Applications](#)

