

Application Setup

Create a Hummingbird application.

Create a simple web application using the Hummingbird template.


5mins

Estimated Time

Section 1

Create your project

Clone the Hummingbird template, configure a project and review the contents of the generated project.

 Clone template

No Preview ↗

```
1 > git clone https://github.com/hummingbird-project/template
```

Clone the Hummingbird template GitHub project

Step 2

Create your project, using the template configure script. Press return on each question to use the default value.

Step 3

Now lets review what the template has setup. Open `Package.swift`.

You can see if has dependencies for Hummingbird and the Apple's Argument Parser library.

Step 4

Open `Sources/App/App.swift`

This contains an `App` type conforming to `AsyncParsableCommand` with three options, the `hostname` and `port` are used to define the server bind address, `logLevel` sets the level of logging required. Finally the `run()` function which calls `buildApplication(_:)` to create an `Application` and then runs it using `runService()`. You can find out more about the argument parser library [here](#).

Step 5

Open `Sources/App/Application+build.swift` to find the `buildApplication(_:)` function.

Here we create a `Logger` with log level set by either the command line argument mentioned above, or the environment variable `LOG_LEVEL`. We then call a function `build`

`Router()` and use the result of that to create our Application.

Step 6

If we look further down the file we can find the `buildRouter()` function.

Here we create the Router. We add a logging middleware to it (this logs all requests to the router). The function uses a result builder to create a stack of middleware, but you can also use `Router.add(middleware:)` to add individual middleware. Finally we add a single endpoint `GET /` which returns a String response: "Hello".

Step 7

We can run this application and use curl to test it works.

Step 8

Now we have a running server, lets add some functionality to it.

Next

Add your application API

Add some functionality to your application.

Get started

