

[Hummingbird](#) / HTTPResponder

Protocol

HTTPResponder

Protocol for object that produces a response given a request

```
protocol HTTPResponder<Context> : Sendable
```

Overview

This is the core protocol for Hummingbird. It defines an object that can respond to a request.

Topics

Associated Types

associatedtype Context

Required

Instance Methods

```
func respond(to: Request, context: Context) async throws -> Response
```

Return response to the request supplied

Required

Relationships

Inherits From

`Swift.Sendable`

Conforming Types

`CallbackResponder`

`RouterBuilder`

Conforms when `Context` conforms to `RouterRequestContext`, `Context` is `Handler.Context`, `Handler` conforms to `MiddlewareProtocol`, `Handler.Input` is `Request`, and `Handler.Output` is `Response`.

`RouterResponder`

See Also

Router

`class Router`

Create rules for routing requests and then create `Responder` that will follow these rules.

`struct RouterGroup`

Used to group together routes under a single path. Additional middleware can be added to the endpoint and each route can add a suffix to the endpoint path

`class RouteCollection`

Collection of routes

`protocol RouterMethods`

Conform to RouterMethods to add standard router verb (get, post ...) methods

`struct RouterOptions`

Router Options

`protocol HTTPResponderBuilder`

A type that has a single method to build a HTTPResponder

`struct CallbackResponder`

Responder that calls supplied closure

`struct RouterResponder`

`struct EndpointPath`

Endpoint path storage

`struct RouterPath`

Split router path into components

`struct RequestID`

Generate Unique ID for each request