Javalin

No not the spear

Javalin

- Lightweight web framework used for creating web applications in Java (and Kotlin)
- Runs on an embedded Jetty server that hosts your program allowing it to handle http requests and send responses
  - Jetty is a java web server and servlet container

Context Object

- The `Context` object provides you with everything you need to handle a http-request.
- It contains the underlying servlet-request and servlet-response, and a bunch of getters and setters.
- The servlet-request and servlet-response objects are the object equivalents of your HTTP request and response
- The underlying servlet container (Jetty) transforms the incoming HTTP request to a servlet request object and the servlet response to the actual HTTP response we send back to the client

Handler

- Interface used to define endpoint actions that your program executes to *handle* your http requests
- Fun fact: it's a functional interface which means you can define implementations of it through lambda expressions

Processing Request & Sending Response Data

- Marshalling & unmarshalling
  - Marshalling is the process of converting Java objects to a format that is transferable over the wire (i.e. the network)
  - Unmarshalling is the process of disassembling data that is in external data representation form, into a locally interpretable form (i.e. Java object)
- Jackson
  - is a very popular and efficient java based library used to serialize or map java objects to JSON and vice versa.
  - Uses an objectMapper class to perform marshalling and unmarshalling

Front Controller Design Pattern

- used to provide a centralized request handling mechanism so that all requests will be handled by a single handler
- This handler will be in charge of routing the requests to the controller methods in charge of processing the requests

Resources

- Javalin Docs
- Front Controller Design Pattern

Questions?

Comments? Concerns? Violent Reactions?