GenServer Basics

Eric Oestrich

smartlogic.io

GenServer

https://hexdocs.pm/elixir/GenServer.html

Basics

- Keeps state via tail recursion
- Uses built in erlang process communication (messages)
- Handles the generic server implementation, error handling, etc.

start_link

- Will start the elixir process, linking it to the current one
- Can use with OTP and an application

init

- Called before returning from start_link
- Can do work transforming the passed in state

GenServer.call and handle_call

- A callback function that is expected to reply
- GenServer.call is synchronous

GenServer.cast and handle_cast

- A callback function that is not expected to reply
- GenServer.cast is asynchronous

handle_info

- This callback is used to handle all other GenServer messages

Design Principles

- Keep the GenServer light, do work in another module
- Have a client API for interacting with the GenServer

OTP - Supervisors

- Handle restarts
- Use the `:observer.start()` tool to inspect