ACTIX CRUD APP

Speaker: Dennoh Peter

x.com/dennohpeter

github.com/dennohpeter

Introduction

- Actix Web: is a powerful, pragmatic, and extremely fast web framework for Rust

- Features
  - Type Safe: from request to response, everything has types
  - Feature Rich: HTTP/2, Logging, Ws
  - Blazingly Fast
  - Extensible: create your own libs that any Actix Application can use

To Cover

- HttpServer
- Application
  - state
  - middleware
  - routes
  - services
- Telemetry/Logging
- Requests
  - handlers
  - extractors

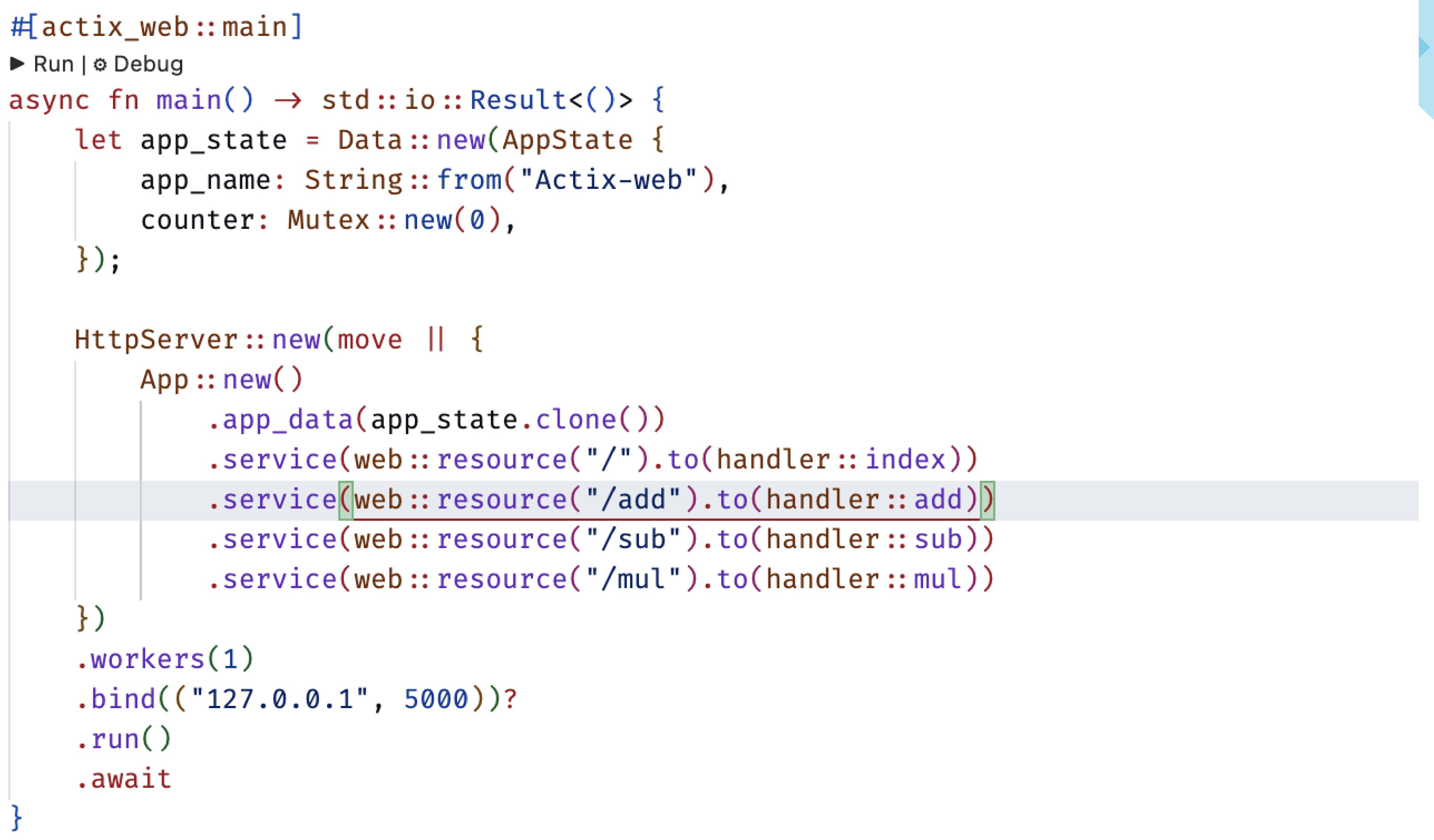
  $\left.\begin{matrix}\\\\\end{matrix}\right\}$ validation
- Response
  - error handling
- Persistence: Databases

Actix: Application

Used to register routes for **resources** and **middleware**

Stores **application STATE shared across all handlers** within the **same scope**

Scope ~= namespace for all routes



```rust
#[actix_web::main]
async fn main() -> std::io::Result<()> {
    let app_state = Data::new(AppState {
        app_name: String::from("Actix-web"),
        counter: Mutex::new(0),
    });

    HttpServer::new(move || {
        App::new()
            .app_data(app_state.clone())
            .service(web::resource("/").to(handler::index))
            .service(web::resource("/add").to(handler::add))
            .service(web::resource("/sub").to(handler::sub))
            .service(web::resource("/mul").to(handler::mul))
    })
    .workers(1)
    .bind(("127.0.0.1", 5000))?
    .run()
    .await
}
```

Actix: Handlers

- Async fn that accepts zero or more params that are **called data extractors** – which provide **type-safe access** to data contained in HTTP requests using ***deserialization*** with `**serde**, and returns a type that can be converted into an HttpResponse(i.e impl Responder)
  - Path
  - Query
  - JSON
  - Data
  - HttpRequest
  - String
  - Bytes
  - Payload

Actix: Requests

- JSON request
- Validation

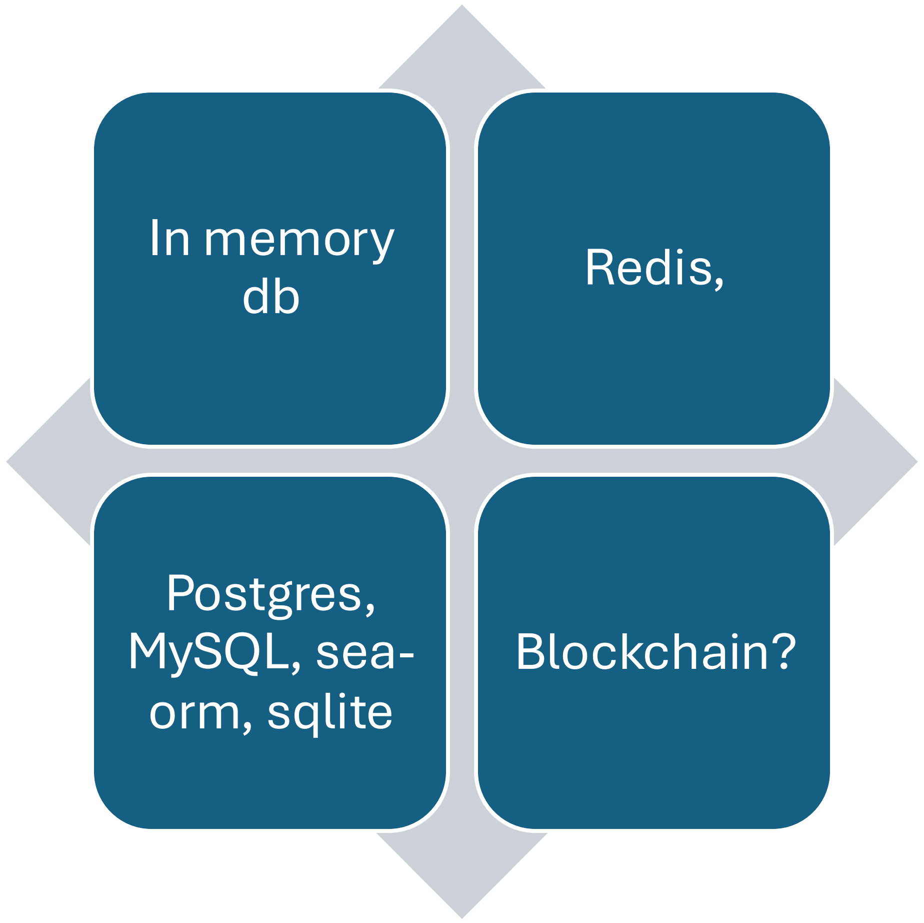
Actix: Error Handling

- thiserror
- anyhow

Actix: HttpResponse

- HttpResponseBuilder methods -> body, finish, json finalize response creation and return a constructed HttpResponse instancre
- JSON Response – well formatted json
- JSON<T> where T is the type of a structure to serialize into JSON. T must impl **Serialize** trait

Databases


In memory db
Redis,
Postgres, MySQL, sea-orm, sqlite
Blockchain?

Telemetry

- Telemetry: What for?
- Crates available
  - Log => env_logger

  More ?
  - Tracing Ecosystem

  Tracing-subscriber => tracing_log => tracing-bunyan-formatter => tracing-actix-web => tracing-error

CRUD App

- Idea suggestion: ?

References

- Actix Web Docs - https://actix.rs
- env_logger - https://docs.rs/env_logger
- log - https://docs.rs/log
- Zero to Production In Rust ~ Luca Palmieri - https://www.zero2prod.com

Resources

Zero to Production In Rust ~ Luca Palmieri

Effective Rust ~ David Drysdale

Polkadot ~ https://docs.polkadot.com/develop

THANK YOU

Connect </>

t.me/dennohpeter

x.com/dennohpeter

github.com/dennohpeter