Introduction to Deno

A secure runtime for JavaScript and TypeScript.

What is it?

From the website: Deno is a simple, modern and secure runtime for JavaScript and TypeScript that uses V8 and is built in Rust.

1. Secure by default. No file, network, or environment access, unless explicitly enabled.
2. **Supports TypeScript out of the box.**
3. Ships only a single executable file.
4. Has **built-in utilities** like a dependency inspector (deno info) and a code formatter (deno fmt).
5. Has a set of reviewed (audited) **standard modules** that are guaranteed to work with Deno: deno.land/std

Why do we care?

1. Sounds like nodejs? Same person who created nodejs created deno!
2. Both Javascript and Typescript can be used outside of the browser. No need to transpile, deno works with typescript.
3. Deno has a standard library.
4. Deno can use web assembly.
5. No package.jsons needed! (most controversial change but more later)

Getting Started - Install and Example

```
shift3@everyday:~$ curl -fsSL https://deno.land/x/install/install.sh | sh
```

```
shift3@everyday:~$ deno run https://deno.land/std/examples/welcome.ts
Download https://deno.land/std/examples/welcome.ts
Warning Implicitly using master branch https://deno.land/std/examples/welcome.ts
Check https://deno.land/std/examples/welcome.ts
Welcome to Deno 🦕
```

Getting Started - Part 2 (Simple web server)

```typescript
import { serve } from "https://deno.land/std@0.61.0/http/server.ts";
const s = serve({ port: 8000 });
console.log("http://localhost:8000/");
for await (const req of s) {
  req.respond({ body: "Hello World\n" });
}
~
```

```
shift3@everyday:~/Documents/Github/deno-developer-talk/examples/intro$ deno run --allow-net intro_world_server.ts
http://localhost:8000/
```

```
← → C ⌂    🛡 ⓘ localhost:8000
```

Hello World

Getting Started - Part 3 (More complicated)

Other example programs (all based on https://deno.land/manual):
- Making an HTTP request
- Reading Files
- TCP server
- An implementation of the unix "cat" program
- File server
- TCP echo server
- Run subprocess
- Inspecting and revoking permissions
- Handle OS Signals
- File system events

Example REST API
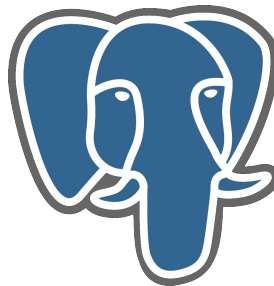
How to build a RESTFUL API with deno?

- Deno (duh)
- OAK: A middleware framework for Deno's net server
- PostgreSQL: The World's Most Advanced Open Source Relational Database

+ **+** **=**

Following guide here: https://github.com/bradtraversy/deno-rest-api)

References and Resources

- Logo from https://github.com/denolib/high-res-deno-logo/blob/master/deno_hr_circle.svg
- Main website: https://deno.land/
- Documentation and example programs: https://deno.land/manual
- FreeCodeCamp more in depth 6 hour talk: https://www.youtube.com/watch?v=TQUy8ENesGY
- Deno crash course with Rest API: https://www.youtube.com/watch?v=NHHhiqwcfRM&t=5s
- Oak REST API: https://github.com/oakserver/oak
- Other Oak REST API: https://medium.com/javascript-in-plain-english/building-crud-apis-using-deno-and-oak-9f71ec106b0e