Node.js

* Runtime environment of JavaScript
  + Ryan Dahl of Google
* It wasn’t the first server side JS environment
  + It was the first to actually be performant.
  + It could hold its own against Java or Python.
* Node.JS uses the V8 JS Engine
  + Same engine that chrome browser uses.
* Very popular backend.
  + Part of the JS everywhere paradigm.
* Node Package Manager (NPM)
  + Think of Maven Repository except for Node.js

TypeScript

* TypeScript is a super set of JavaScript.
  + ANY JS is also valid TS.
  + TS has additional features.
  + JavaScript that Scales
* Statically but not strongly typed language
  + You can optionally use types
  + Values can still be coerced and it is not enforceable
* Origins of TypeScript
  + JS has a lot of great (terrible) design choices
  + The dynamic coercive nature of JS lends itself to errors
    - Objects without the properties you are looking for.
    - Calling functions with incorrect argument types of parameter.
    - Not knowing what type you returned from a function
    - Accidentally making things global scope
  + This part of JS made it very difficult to scale to large applications.
  + Microsoft developers really liked JS and wanted to create a version that would scale better.
* The core feature of **Type**Script is that it has **Typing**.
* TS itself cannot be run.
  + Transpiled into JS to actually be executed.
* VS Code is written in TS.
* TS beat Python for most loved language. Rust is still 1st.