TypeScript!

TypeScript!

JavaScript that scales.

TypeScript is a typed superset of JavaScript that compiles to plain JavaScript.

TypeSecript

```
// JavaScript, types are implicit
const add = (a, b) => a + b
```

TypeSecript

```
// JavaScript, types are implicit
const add = (a, b) => a + b

// TypeScript, types are explicit
const add = (a: number, b: number): number => a + b
```

Why Types?

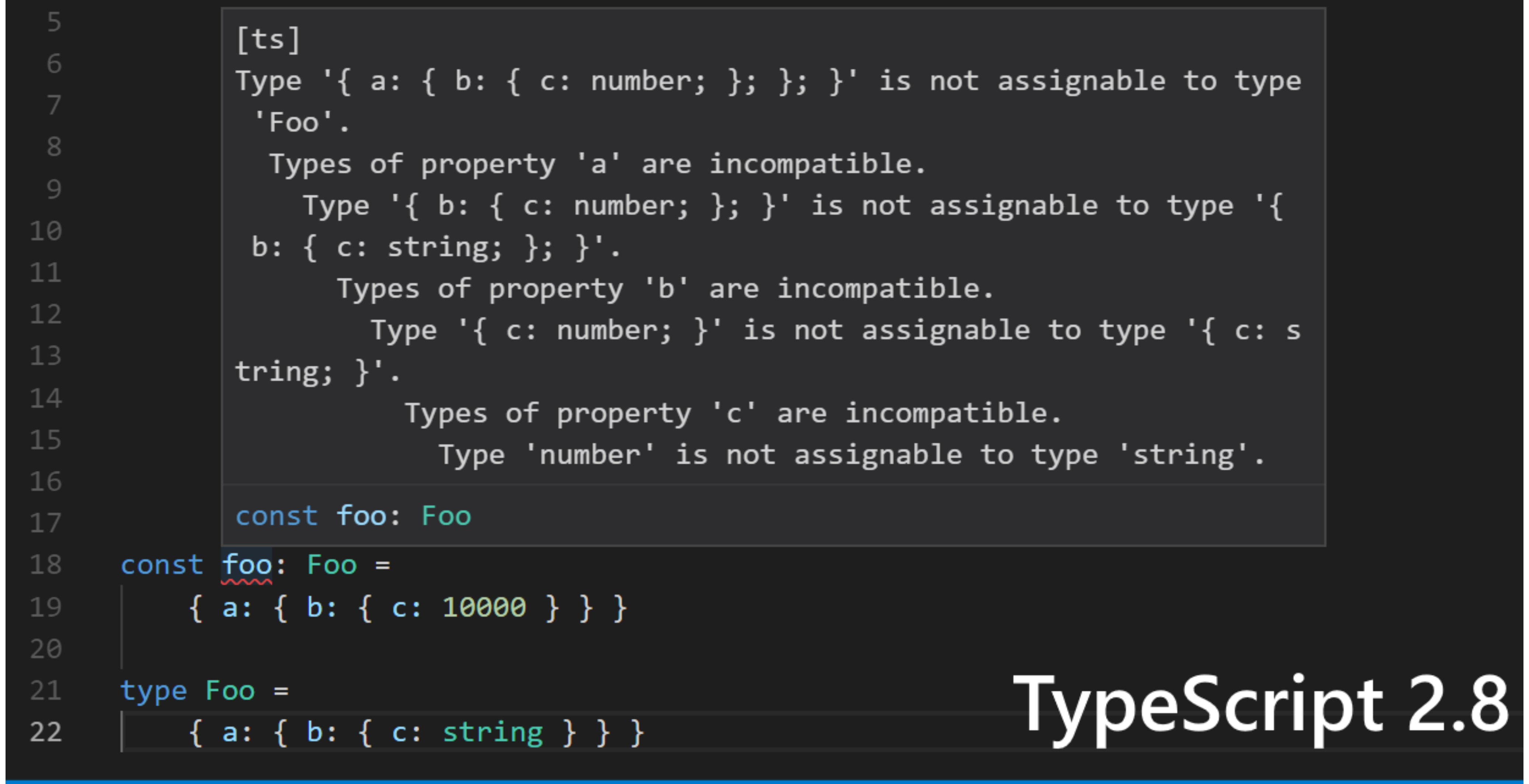
- Catch bugs
- Catch bugs (earlier)
- Documentation & productivity
- GraphQL has a type system
- Refactoring

Why Types?

- Catch bugs
- Catch bugs (earlier)
- Documentation / improve productivity
- GraphQL has a type system
- **Refactoring**

Java?

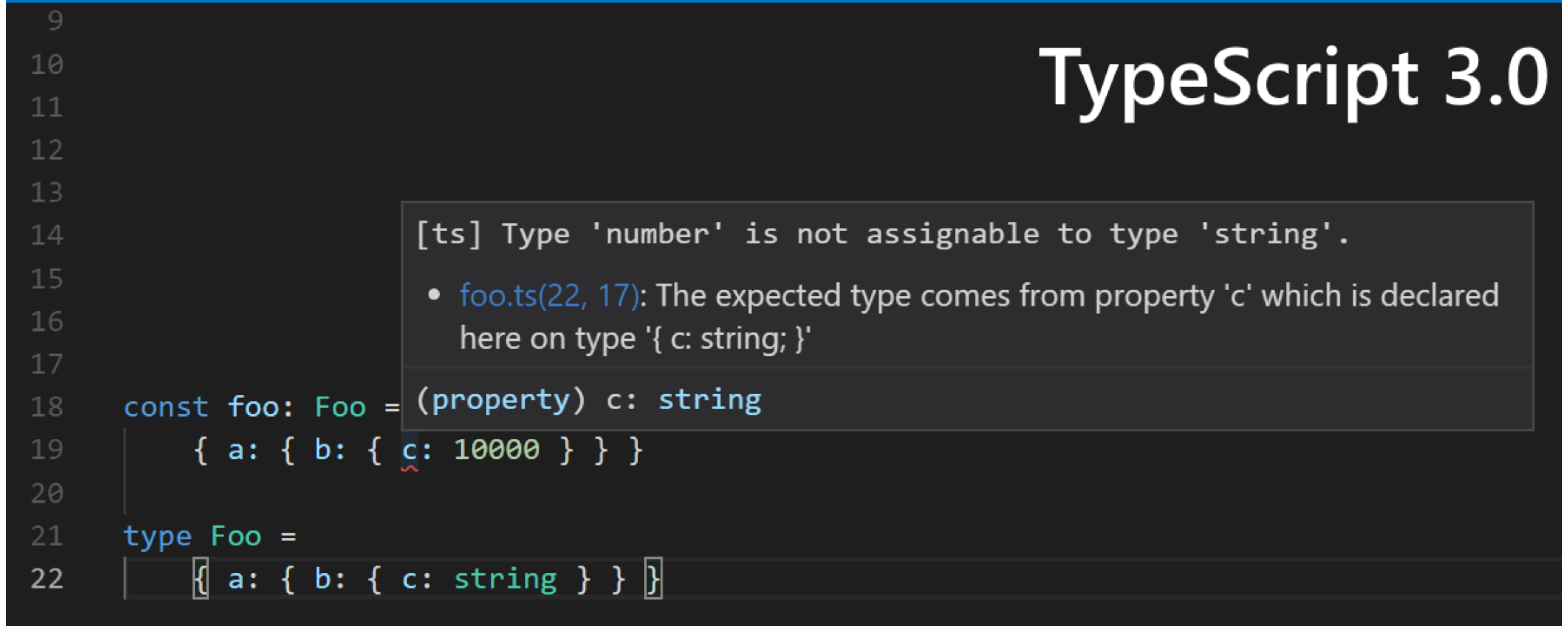
```java
public class Main {
    public static void main(String[] args) {
        int number1 = 10;
        String message = new String("Hello World! ");
        System.out.println(message + number1);
    }
}
```



```
[ts]
Type '{ a: { b: { c: number; }; }; }' is not assignable to type
 'Foo'.
  Types of property 'a' are incompatible.
    Type '{ b: { c: number; }; }' is not assignable to type '{
 b: { c: string; }; }'.
      Types of property 'b' are incompatible.
        Type '{ c: number; }' is not assignable to type '{ c: s
tring; }'.
          Types of property 'c' are incompatible.
            Type 'number' is not assignable to type 'string'.
const foo: Foo
```

```
18  const foo: Foo =
19      { a: { b: { c: 10000 } } }
20
21  type Foo =
22      { a: { b: { c: string } } }
```

TypeScript 2.8



TypeScript 3.0

```
[ts] Type 'number' is not assignable to type 'string'.
```

- foo.ts(22, 17): The expected type comes from property 'c' which is declared here on type '{ c: string; }'

```
(property) c: string
```

```
18  const foo: Foo =
19      { a: { b: { c: 10000 } } }
20
21  type Foo =
22      { a: { b: { c: string } } }
```

Types are all the rage

typing — Support for type hints ¶

New in version 3.5.

Source code: Lib/typing.py

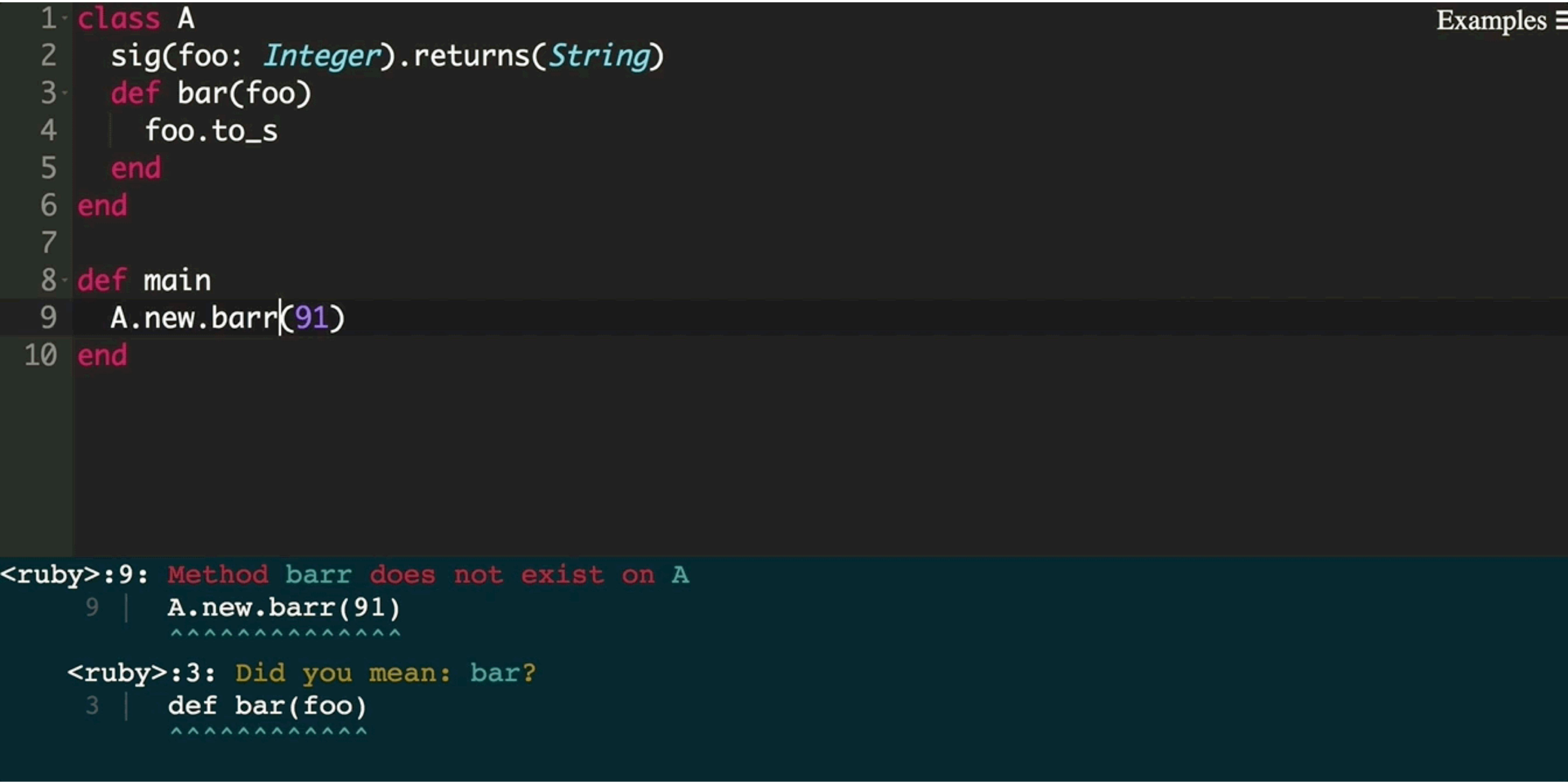
Note: The typing module has been included in the standard library on a provisional basis. New features might be added and API may change even between minor releases if deemed necessary by the core developers.

This module supports type hints as specified by **PEP 484** and **PEP 526**. The most fundamental support consists of the types `Any`, `Union`, `Tuple`, `Callable`, `TypeVar`, and `Generic`. For full specification please see **PEP 484**. For a simplified introduction to type hints see **PEP 483**.

The function below takes and returns a string and is annotated as follows:

```python
def greeting(name: str) -> str:
    return 'Hello ' + name
```

In the function `greeting`, the argument `name` is expected to be of type `str` and the return type `str`. Subtypes are accepted as arguments.


Examples ☰

```ruby
class A
  sig(foo: Integer).returns(String)
  def bar(foo)
    foo.to_s
  end
end

def main
  A.new.barr(91)
end
```

```
<ruby>:9: Method barr does not exist on A
     9 |   A.new.barr(91)
           ^^^^^^^^^^^^^^

    <ruby>:3: Did you mean: bar?
     3 |   def bar(foo)
           ^^^^^^^^^^^^
```

Matz: Yeah
the third major goal of the Ruby 3 is adding some kind of static
typing while keeping the duck typing
so some kind of structure for soft-typing or something like that.

-- Ruby 3x3: Matz Koichi and Tenderlove on the future of Ruby Performance

TypeScript

PURESCRIPT

flow

REASON

Scala.js

elm

Why TypeScript?

TypeScript

PURESCRIPT

flow

REASON

Scala.js

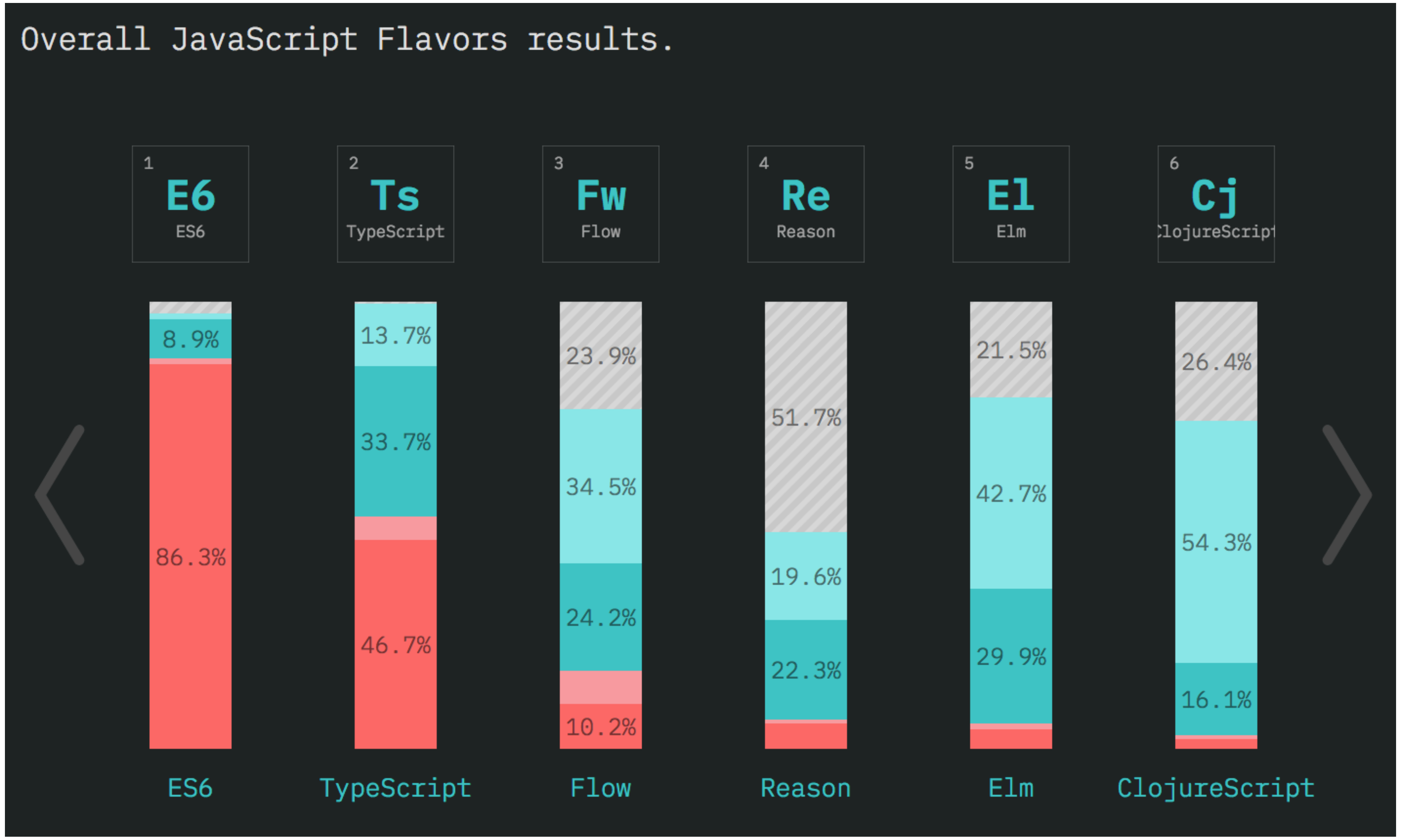
elm

Gradual Adoption Story

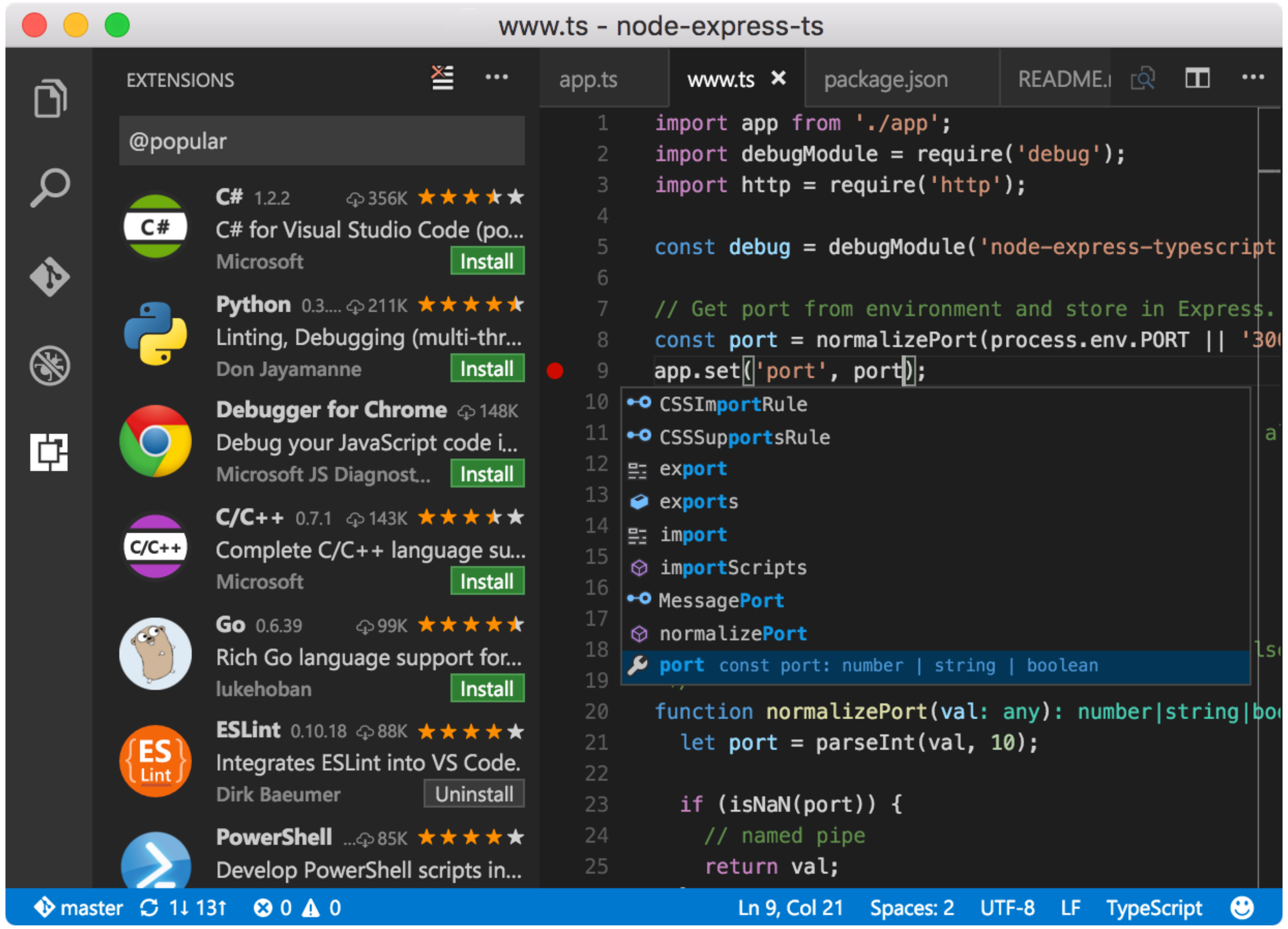
- All JS is valid TS!
- Rename `*.js` -> `*.ts{x}`
- Introduce types for libraries
- Dial up the strictness (towards `--strict` mode)

```
--noImplicitAny

--strictNullChecks

--strictFunctionTypes
```

Community Adoption



Tooling!



www.ts - node-express-ts

EXTENSIONS

@popular

C# 1.2.2	356K	★★★★★	C# for Visual Studio Code (po...	Microsoft · Install
Python 0.3....	211K	★★★★★	Linting, Debugging (multi-thr...	Don Jayamanne · Install
Debugger for Chrome	148K		Debug your JavaScript code i...	Microsoft JS Diagnost... · Install
C/C++ 0.7.1	143K	★★★★★	Complete C/C++ language su...	Microsoft · Install
Go 0.6.39	99K	★★★★★	Rich Go language support for...	lukehoban · Install
ESLint 0.10.18	88K	★★★★★	Integrates ESLint into VS Code.	Dirk Baeumer · Uninstall
PowerShell ...	85K	★★★★★	Develop PowerShell scripts in...	

app.ts | www.ts ✕ | package.json | README.

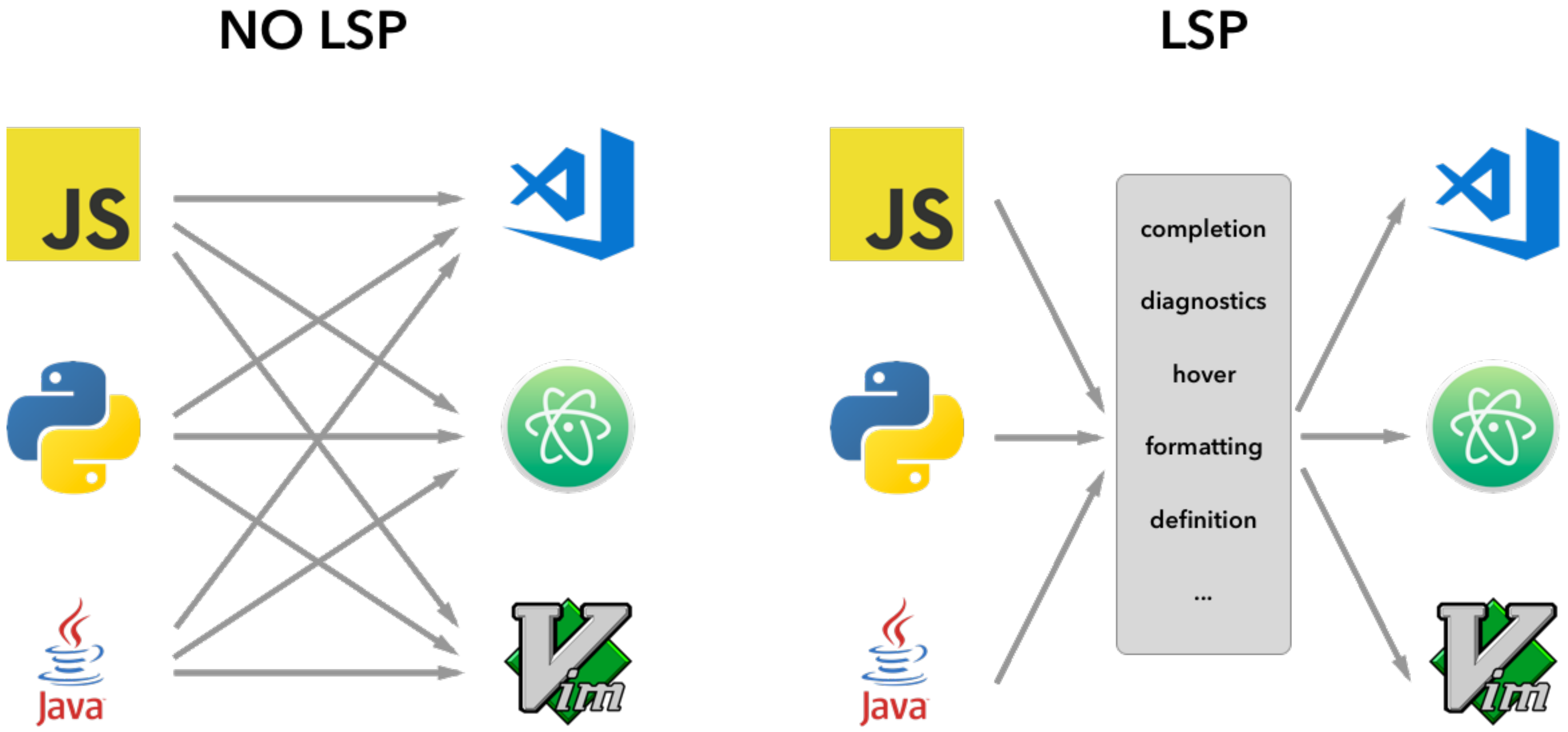
```
import app from './app';
import debugModule = require('debug');
import http = require('http');

const debug = debugModule('node-express-typescript

// Get port from environment and store in Express.
const port = normalizePort(process.env.PORT || '30
app.set('port', port);
```

```
CSSImportRule
CSSSupportsRule
export
exports
import
importScripts
MessagePort
normalizePort
port  const port: number | string | boolean
```

```
function normalizePort(val: any): number|string|boo
  let port = parseInt(val, 10);

  if (isNaN(port)) {
    // named pipe
    return val;
```

master 1↓ 13↑ ⊗ 0 ⚠ 0 Ln 9, Col 21 Spaces: 2 UTF-8 LF TypeScript

Language Server



Library Support

- Apollo & formik are authored in TypeScript
- React very well supported

How to get started

- All JS is valid TS!
- Rename `*.js` -> `*.ts{x}`
- Introduce types for libraries
- Dial up the strictness (towards `--strict` mode)

```
--noImplicitAny

--strictNullChecks

--strictFunctionTypes
```