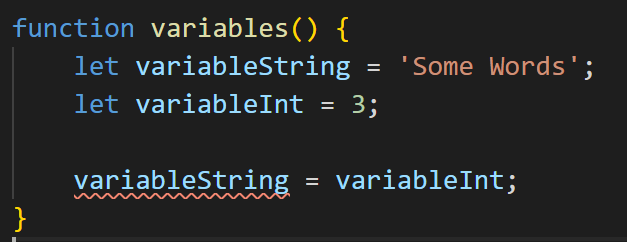
**Preventing JavaScript woes with TypeScript Types**

JavaScript has been a staple for frontend development community since its announcement on December 4th, 1995. To use a quote from Uncle Ben in Spider-Man “With great power comes great responsibility”. The importance that JS has because of today’s internet usage has stretched its power to the breaking point leading to many developer woes as Java Script’s flexibility hides many errors programmers make. In 2012, TypeScript arrived to save the day by introducing tools to help identify and prevent errors before runtime.

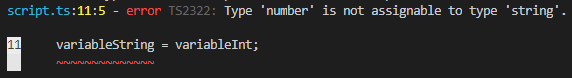
I’ll share with you a couple of basic comparisons between JS and TS that will demonstrate how TS can improve your frontend development.

**Typed Variable**

TS prevents the wrong value type from being assigned to your variables.

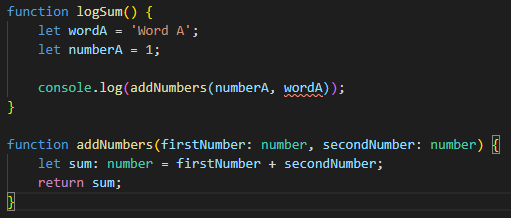


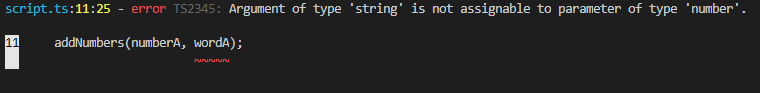
In this example TS throws an error when you assign variableInt to variableString.



JS allows this operation, which can expose possible errors in runtime value assignment.

**Typed Params**

TS allows you to also control the type of parameters that your functions allows. 



In this example, ‘wordA’ and ‘numberA’ are passed into a function that will add them and come out with a sum. In JS you would not be warned that you are passing a word where a number would be; TS will show you an error because ‘addNumbers’ parameters are of type “number”.

**Typed Classes**

TS allows you to specify custom types to use in your code and keeps them clear of the incorrect data.

