Cheatsheets / **Learn TypeScript**

Types

TypeScript

The TypeScript programming language is a superset of JavaScript that adds types to JavaScript using a set of tools called a type system.

Primitive Types

All “primitive”, or built-in data types in JavaScript are recognized by TypeScript.

```
"Hello, world!"  // string
42               // number
true             // boolean
null
undefined
```

TypeScript Type Inference

Type inference assumes the expected type of the variable throughout a program based on the data type of the value initially assigned to it at declaration. Type inference will log a complaint if the variable is later reassigned to a different type.

```
let first = 'Anders';

first = 1337; // Type 'number' is not
assignable to type 'string'
```

The Shape of an Object

TypeScript knows the *shape* of an object—what member properties it does or doesn’t contain. TypeScript will log an error if the code attempts to access members of an object known not to exist. It may even suggest possible corrections.

```
let firstName = 'muriel!';

console.log(firstName.toUppercase());

// error: Property 'toUppercase' does
not exist on type 'string'. Did you
mean 'toUpperCase'?
```

Typescript any

When a variable is declared without being assigned an initial value, TypeScript considers it to be of type `any`. A variable of this type can be reassigned without generating any error from TypeScript.

```
let first;
first = 'Anders';
first = 1337;
```

TypeScript Supports Type Annotations

Adding a type annotation ensures that a variable can only ever be assigned to that type. This can be useful when declaring a variable without an initial value. Type annotations get removed when compiled to JavaScript. The type declaration is provided by appending a variable with a colon (`:`) and the type (eg. `number`).

```
let first: string;
first = 'Anders';

// Error: Type 'number' is not
assignable to type 'string'
first = 1337;
```