

PROGRAMMING ON THE TOILET

What is it?

Programming on the Toilet is a document which contains tips, tricks, snippets or explanations about various programming technologies and paradigms, which could help developers in their everyday job. This episode was brought to you by Kenan Klisura, Lead Software Engineer at Atlantbh.

Falsy and truthy values in Javascript

Falsy values in javascript are: false, null, undefined, 0, NaN, "" (empty strings). The following construct:

```
let val = ...;
if (!val) {
  // Execute
}
```

... will always Execute if the val contains one of the above values.

Truthy values are all others. Some examples are: {} (empty object), [] (empty array), function () {} (empty function), '0' (string with 0 as value). This enables us to write:

```
if (val) {
  // We're guaranteed val will not be null, undefined or any other
  // falsy value.
}
```

Use this when writing if statements, instead of writing val == null or val == undefined.

Arrow functions in Javascript

Arrow functions were introduced in ES6 and represent a shorter syntax for function declaration. They don't have their own this (or arguments) and they preserve this context of the outer function. The following examples will print test on the output if the fn function is called on object. Observe the console.log argument in the example below, with and without arrow function.

<i>Without arrow function</i>	<i>With arrow function.</i>
<pre>{ test: 'test', fn: function() { const that = this; function sub() { console.log(that.test); } sub(); } }</pre>	<pre>{ test: 'test', fn: function() { const sub = () => console.log(this.test); sub(); } }</pre>

Use the arrow functions on the reduce, map, forEach, filter and similar functions and only if the function body is short.