



# The Angular-Arrow functions cheatsheet

## The Must-Have Rules

- Never use arrow functions for `module.service('MyService', ...)`
- Never use arrow functions when defining controllers, e.g. `controller: () => {}` inside a component

## The recommended style guide

- Never use arrow functions as parameters to any `module.foo()` methods
- Never use arrow functions when defining a controller

*\*Also keep in mind there are specific libraries that don't play well with arrow functions specifically, most notably Mocha.*

## The Syntax Sugar

### Regular function

```
(arg1, arg2) => {  
  // do stuff  
  return arg1 + arg2;  
}
```

### Implicit Return

If your function only returns a single expression, you don't have to use braces:

```
(arg1, arg2) => arg1 + arg2
```

### Single Argument

If you only have a single argument, the parens are optional:

```
arg => arg * 2
```

### No Arguments

And if you have no arguments the syntax goes like:

```
() => {  
  // do stuff  
}
```

*\*You might see people use `_ => {}`. It shadows the common variable name for lodash/underscore, so I recommend against it.*

### Returning an object literal

This **won't** work:

```
x => {foo: x}
```

You have to wrap the object literal with parens:

```
x => ({foo: x})
```