

Twitter Thread by Insha



Insha

[@Insharamin](#)



Being developers, we work with many different languages.

Which makes it humanly impossible to remember all its aspects.

This is when a "Cheat Sheet" comes super-handy.

Here's the ultimate list of Cheatsheets for Web Developers■■■↓

In this post, we'll go through some quick and useful cheat sheets that'll save you hours of wasted time each week.

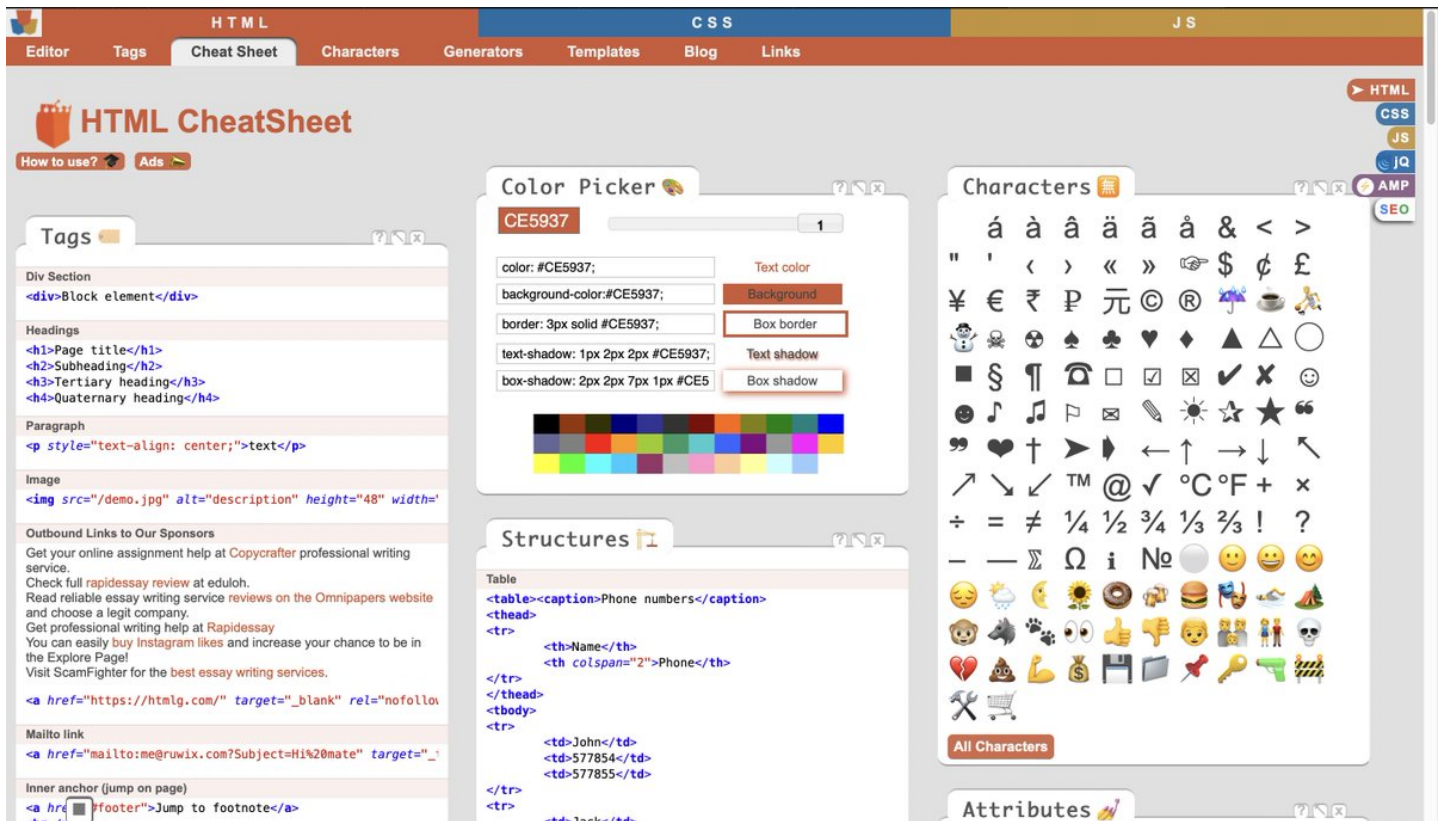
When it comes to Web development, HTML is the first choice. Web development generally comprises 3 fundamentals: basically HTML, CSS, and JAVASCRIPT.

■ HTML Cheat Sheet

Looking for a cheat sheet that contains useful code examples and web developer tools, markup generators, and more? Well, you're going to love this.

It covers all the basic syntax along with proper examples.

■ <https://t.co/VwnLHVT8BV>



■ HTML Reference

If you believe in learning by examples: Let me tell you about this one.

It is a free guide to HTML. You will find all the elements and attributes with proper examples for a better understanding.

■ <https://t.co/l9ldkR5qT6>

Search 113 HTML elements

☒ experimental ☒ meta ☒ self-closing ☒ inline ☒ block

a	meta	self-closing	inline	block
abbr	meta	self-closing	inline	block
address	meta	self-closing	inline	block
area	meta	self-closing	inline	block
article	meta	self-closing	inline	block
aside	meta	self-closing	inline	block
audio	meta	self-closing	inline	block
b	meta	self-closing	inline	block
base	meta	self-closing	inline	block
bdi	meta	self-closing	inline	block
bdo	meta	self-closing	inline	block
blockquote	meta	self-closing	inline	block
body	meta	self-closing	inline	block
br	meta	self-closing	inline	block
button	meta	self-closing	inline	block
canvas	meta	self-closing	inline	block
caption	meta	self-closing	inline	block
cite	meta	self-closing	inline	block
code	meta	self-closing	inline	block
col	meta	self-closing	inline	block

■ CSS Reference

As we know CSS has a number of properties.

Do you remember them all? I guess nobody does. So what's the hack? These cheat sheets are a hack.

■ <https://t.co/3ohDkiC1ir>

🔍 Search 129 CSS properties	
align-content	flexbox
align-items	flexbox
align-self	flexbox
animation-delay	animations
animation-direction	animations
animation-duration	animations
animation-fill-mode	animations
animation-iteration-count	animations
animation-name	animations
animation-play-state	animations
animation-timing-function	animations
animation	animations
background-attachment	backgrounds
background-clip	backgrounds
background-color	backgrounds
background-image	backgrounds
background-origin	backgrounds
background-position	backgrounds
background-repeat	backgrounds
background-size	backgrounds
background	backgrounds

■ CSS Cheat Sheet

CSS Cheat Sheet contains the most common style snippets: CSS gradient, background, button, font family, border, radius, box, text-shadow generators, color picker, and more.

■ <https://t.co/sUE9Za1bHA>

CSS CheatSheet

How to use?

Properties

align-content: behavior of the flex-wrap property

align-items: alignment for items inside the container

align-self: alignment for the selected item

all: changes all properties

animation: binds an animation to an element

animation-delay: delays animation start

animation-direction: reverse animation or in alternate cycles

animation-duration: animation duration in seconds or ms

animation-fill-mode: style when the animation is not playing

animation-iteration-count: number of an animation replays

animation-name: name for the @keyframes animation

Color Picker

Background

Basics

Syntax

```
selector{
  property: value;
  property2: value2;
}
```

Include external css file

```
<link rel="stylesheet" type="text/css" href="/style.css" .
```

Internal styles

```
<style type="text/css">
div { color: #444;}
</style>
```

Inline styles

```
<tag style="property: value"> </tag>
```

Clearfix

```
.clearfix:after {
  clear: both;
  content: " ";
  display: block;
  font-size: 0;
  height: 0;
  visibility: hidden;
}
.clearfix { display: inline-block; }
* html .clearfix { height: 1%; }
.clearfix { display: block; }
```

Box model

Selectors

*

div

div.p

div p

div > p

div + p

div ~ p

.classname

#idname

div.classname

div#idname

#idname *

Pseudo classes

a:link

a:active

a:hover

a:visited

p::after(content:"yo");

p::before

input:checked

input:disabled

input:enabled

input:focus

input:in-range

Attribute selectors

a[target]

a[target="_blank"]

[title="chair"]

[class^="chair"]

[class="chair"]

[class\$="chair"]

input[type="button"]

Ads

■ GRID

Learn all about the properties available in the Grid Layout through simple visual examples. I as a developer use this handy cheat sheet all the time.

■ <https://t.co/1ws4qycjVo>

GRID

container

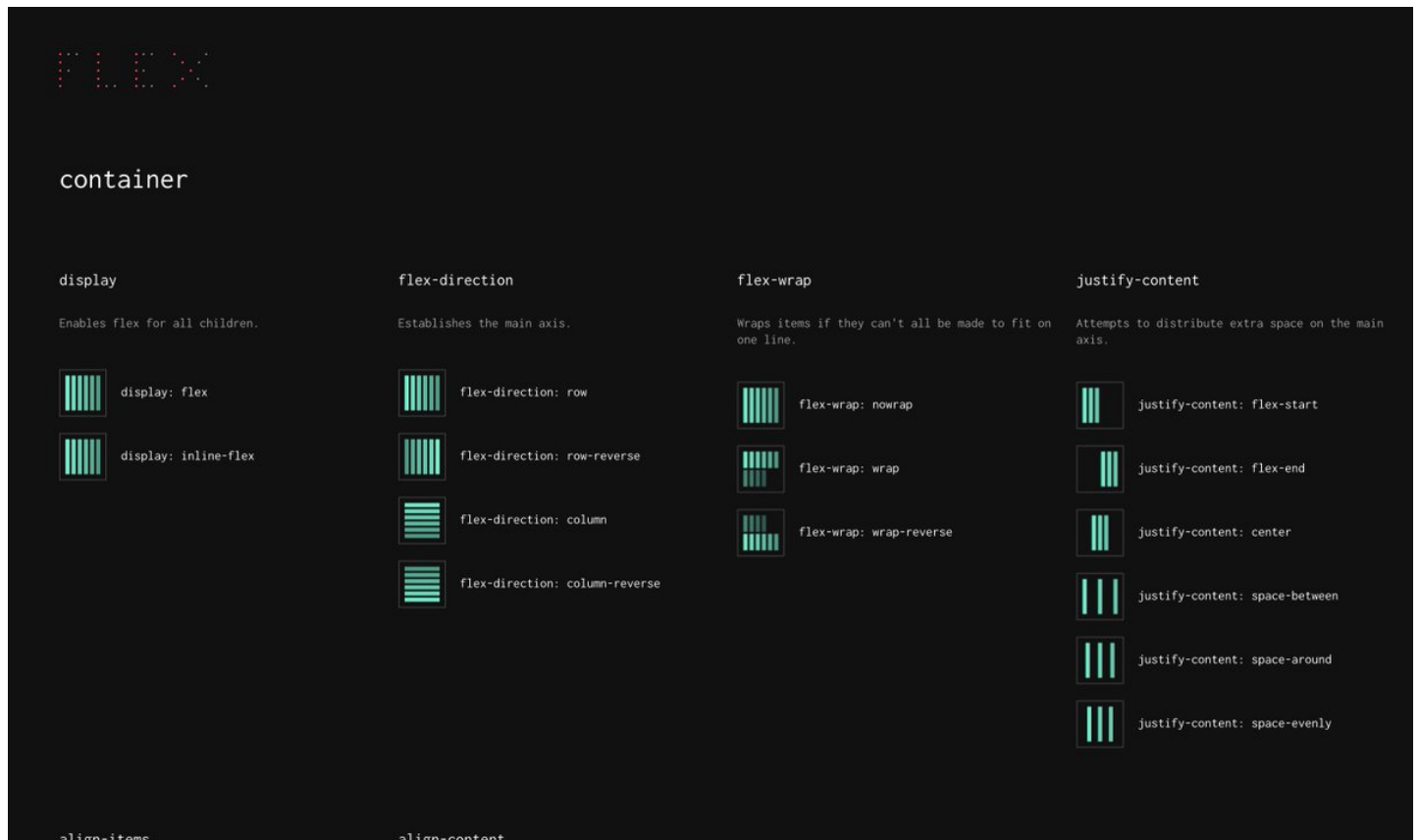
<h3>display</h3> <p>Establishes a new grid formatting context for children.</p> <ul style="list-style-type: none"> display: grid; display: inline-grid; display: subgrid; 	<h3>grid-template</h3> <p>Defines the rows and columns of the grid.</p> <ul style="list-style-type: none"> grid-template-columns: 12px 12px 12px; grid-template-rows: 12px 12px 12px; grid-template-columns: repeat(3, 12px); grid-template-rows: repeat(3, auto); grid-template-columns: 8px auto 8px; grid-template-rows: 8px auto 12px; grid-template-columns: 22% 22% auto; grid-template-rows: 22% auto 22%; 	<h3>grid-gap</h3> <p>Specifies the size of column and row gutters.</p> <ul style="list-style-type: none"> grid-row-gap: 1px; grid-column-gap: 9px; grid-gap: 1px 9px; grid-gap: 6px; 	<h3>justify-items</h3> <p>Aligns content in a grid item along the row axis.</p> <ul style="list-style-type: none"> justify-items: start; justify-items: end; justify-items: center; justify-items: stretch; (default)
<h3>align-items</h3> <p>Aligns content in a grid item along the column axis.</p> <ul style="list-style-type: none"> align-items: start; 	<h3>justify-content</h3> <p>Justifies all grid content on row axis when total grid size is smaller than container.</p> <ul style="list-style-type: none"> justify-content: start; 	<h3>align-content</h3> <p>Justifies all grid content on column axis when total grid size is smaller than container.</p> <ul style="list-style-type: none"> align-content: start; 	<h3>grid-auto-flow</h3> <p>Algorithm for automatically placing grid items that aren't explicitly placed.</p> <ul style="list-style-type: none"> grid-auto-flow: row;

■ FLEX

Learn all about the properties available in flexbox through simple visual examples.

You need to fix the alignment of your webpage but you don't know which property would work? Take a look at these cheat sheets.

■ <https://t.co/VIP2ICisRP>



■ JavaScript Cheatsheet

This 13-page cheat sheet will take you from a beginner to advanced on the fundamentals of JavaScript.

Giving you proper terminologies, and syntax of common constructs.

■ <https://t.co/Y6BZqRGRBO>

① Seven (7) Types

Six Primitive Types

1. String
`"Any text"`
2. Number
`123.45`
3. Boolean
`true` or `false`
4. Null
`null`
5. Undefined
`undefined`
6. Symbol
`Symbol('something')`
7. Object
 - Array
`[1, "text", false]`
 - Function
`function name() { }`

② Basic Vocabulary

Variable

A named reference to a value is a variable.

Operator

Operators are reserved-words that perform action on values and variables. Examples: `+` `-` `*` `in` `===` `typeof` `!=` ...

Statement
A group of words, numbers and operators that **do a task** is a statement.

Note: `var`, `let` & `const` are all valid keywords to declare variables. The difference between them is covered on page 7 of this cheatsheet.

Keyword / reserved word

Any word that is part of the vocabulary of the programming language is called a keyword (a.k.a reserved word). Examples: `var` `=` `+` `if` `for`...

Expression

A reference, value or a group of reference(s) and value(s) combined with operator(s), which result in a single value.

③ Object

An object is a data type in JavaScript that is used to store a combination of data in a simple key-value pair. That's it.

Key

These are the keys in `user` object.

```
var user = {
  name: "Aziz Ali",
  yearOfBirth: 1988,
  calculateAge: function(){
    // some code to calculate age
  }
}
```

Value

These are the values of the respective keys in `user` object.

Method

If a key has a function as a value, it's called a method.

"Don't just learn JavaScript - Become a Full-Stack JavaScript Developer"

 iLoveCoding
<https://iLoveCoding.org>

■ JavaScript (JS) Cheat Sheet

This JavaScript cheat sheet comes with the most important concepts, functions, methods, and more.

A complete quick reference for beginners.

■ <https://t.co/EmTVYHaKdN>

JS CheatSheet

[Ads](#) [Ads](#)

[HTML](#)
[CSS](#)
[JS](#)
[JQ](#)
[AMP](#)
[SEO](#)

If - Else

```
if ((age >= 14) && (age < 19)) { // logical condit:
    status = "Eligible.";        // executed if condit:
} else {                        // else block is e
    status = "Not eligible.";    // executed if condit:
}
```

Switch Statement

```
switch (new Date().getDay()) { // input is current da
case 6:                       // if (day == 6)
    text = "Saturday";
    break;
case 0:                       // if (day == 0)
    text = "Sunday";
    break;
default:                      // else...
    text = "Whatever";
}
```

Variables x

```
var a; // variable
var b = "init"; // string
var c = "Hi" + " " + "Joe"; // = "Hi Joe"
var d = 1 + 2 + "3"; // = "33"
var e = [2,3,5,8]; // array
var f = false; // boolean
var g = /(); // RegEx
var h = function(){}; // function object
const PI = 3.14; // constant
var a = 1, b = 2, c = a + b; // one line
let z = 'zzz'; // block scope local variab
```

Strict mode

```
"use strict"; // Use strict mode to write secure code
```

Basics >

On page script

```
<script type="text/javascript"> ...
</script>
```

Include external JS file

```
<script src="filename.js"></script>
```

Delay - 1 second timeout

```
setTimeout(function () {
}, 1000);
```

Functions

```
function addNumbers(a, b) {
return a + b; ;
}
x = addNumbers(1, 2);
```

Edit DOM element

```
document.getElementById("elementID").innerHTML = "Hello W
```

Output

```
console.log(a); // write to the browser console
document.write(a); // write to the HTML
alert(a); // output in an alert box
confirm("Really?"); // yes/no dialog, returns true/false
prompt("Your age?", "0"); // input dialog, Second argument is optional
```

Comments

```
/* Multi line
comment */
// One line
```

Loops <

For Loop

```
for (var i = 0; i < 10; i++) {
    document.write(i + ": " + i*3 + "<br />");
}
var sum = 0;
for (var i = 0; i < a.length; i++) {
    sum += a[i]; // parsing an array
}
html = "";
for (var i of custOrder) {
    html += "<li>" + i + "</li>";
}
```

While Loop

```
var i = 1; // initialize
while (i < 100) { // enters the cycle if str
    i += 2; // increment to avoid infinity
    document.write(i + ", "); // output
}
```

Do While Loop

```
var i = 1; // initialize
do { // enters cycle at least o
    i += 2; // increment to avoid infinity
    document.write(i + ", "); // output
} while (i < 100) // repeats cycle if statem
```

Break

```
for (var i = 0; i < 10; i++) {
    if (i == 5) { break; } // stops and exits the cycl
    document.write(i + ", "); // last output number is i
}
```

Continue

```
for (var i = 0; i < 10; i++) {
    if (i == 5) { continue; } // skips the rest of the i
    document.write(i + ", "); // skips 5
}
```

Data Types

[HTML](#)
[CSS](#)
[JS](#)
[JQ](#)
[AMP](#)
[SEO](#)

■ React.js cheatsheet

Looking for a quick reference to React.js? Have a look at this cheat sheet.
This guide targets React v15 to v16.

■ <https://t.co/UVdqyV8r2l>

←

DEVHINTS.IO

Edit

React.js cheatsheet

React is a JavaScript library for building user interfaces. This guide targets React v15 to v16.

Components

```
import React from 'react'
import ReactDOM from 'react-dom'

class Hello extends React.Component {
  render () {
    return <div className='message-box'>
      Hello {this.props.name}
    </div>
  }
}

const el = document.body
ReactDOM.render(<Hello name='John' />, el)

Use the React.js jsfiddle to start hacking. (or the unofficial jsbin)
```

Children

```
<AlertBox>
  <h1>You have pending notifications</h1>
</AlertBox>
```

Import multiple exports

```
import React, {Component} from 'react'
import ReactDOM from 'react-dom'

class Hello extends Component {
  ...
}
```

States

```
constructor(props) {
  super(props)
  this.state = { username: undefined }
}

this.setState({ username: 'rstacruz' })

render () {
  this.state.username
  const { username } = this.state
  ...
}
```

Properties

```
<Video fullscreen={true} autoplay={false} />


render () {
  this.props.fullscreen
  const { fullscreen, autoplay } = this.props
  ...
}
```

Use this.props to access properties passed to the component.
See: Properties

Nesting

```
class Info extends Component {
  render () {
    const { avatar, username } = this.props

    return <div>
      <UserAvatar src={avatar} />
      <UserProfile username={username} />
    </div>
  }
}
```



Give your team the onboarding experience they deserve.
Download the guide to learn more.
ads via Carbon

■ <https://t.co/QuRNFAizEj>

Cheatography has thousands of incredible, original programming cheat sheets. You will find everything related to a particular programming language and can easily download them.

■ <https://t.co/uw2tQ4CldR>

Cheatography

e.g., Regular Expressions, Garden, Kitchen, Audio

Home

Cheat Sheets

Create

Community

Help

LOGIN or REGISTER

Programming Cheat Sheets

1719 Programming Cheat Sheets

Sort: Magic

Filter:

Language(s):

Rating:

Cheatography

Regular Expressions Cheat Sheet

1 Page

★★★★★ (371)

Regular Expressions Cheat Sheet

A quick reference guide for regular expressions (regex), including symbols, ranges, grouping, assertions and some sample patterns to get you started.

DaveChild

19 Oct 11, updated 12 Mar 20

development, regex, regular, expressions, programming

Top Tags in Programming

- Python (149)
- Development (109)
- Programming (90)
- JavaScript (69)
- Web (63)
- Git (60)
- HTML (58)
- Network (46)
- Selenium (42)
- Java (42)

Languages

- ASP
- Assembly
- C
- CoffeeScript
- ColdFusion
- Golang
- Haskell
- Java

Cheatography

AngularJS Cheat Sheet

1 Page

★★★★★ (371)

AngularJS Cheat Sheet

The official - unofficial AngularJS cheatsheet.

ProLoser

9 Aug 12, updated 29 Feb 20

■ <https://t.co/7IFOd89iIH>

This website aggregates cheat sheets for many popular programming languages such as JavaScript, NodeJS, PHP, Python, and CSS.

■ <https://t.co/sPrPoS3YMU>

OverAPI.com

Collecting All Cheat Sheets

Python

jQuery

node

php

JS

/*AS*/

CSS

MySQL

RegEx

<HTML>

<HTML>

express

Developer Tools >>

About and Comments >>

Get Chrome Extension >>

A

B

C

D

E

F

G

H

I

J

K

L

M

N

O

P

Q

R

S

T

U

V

W

X

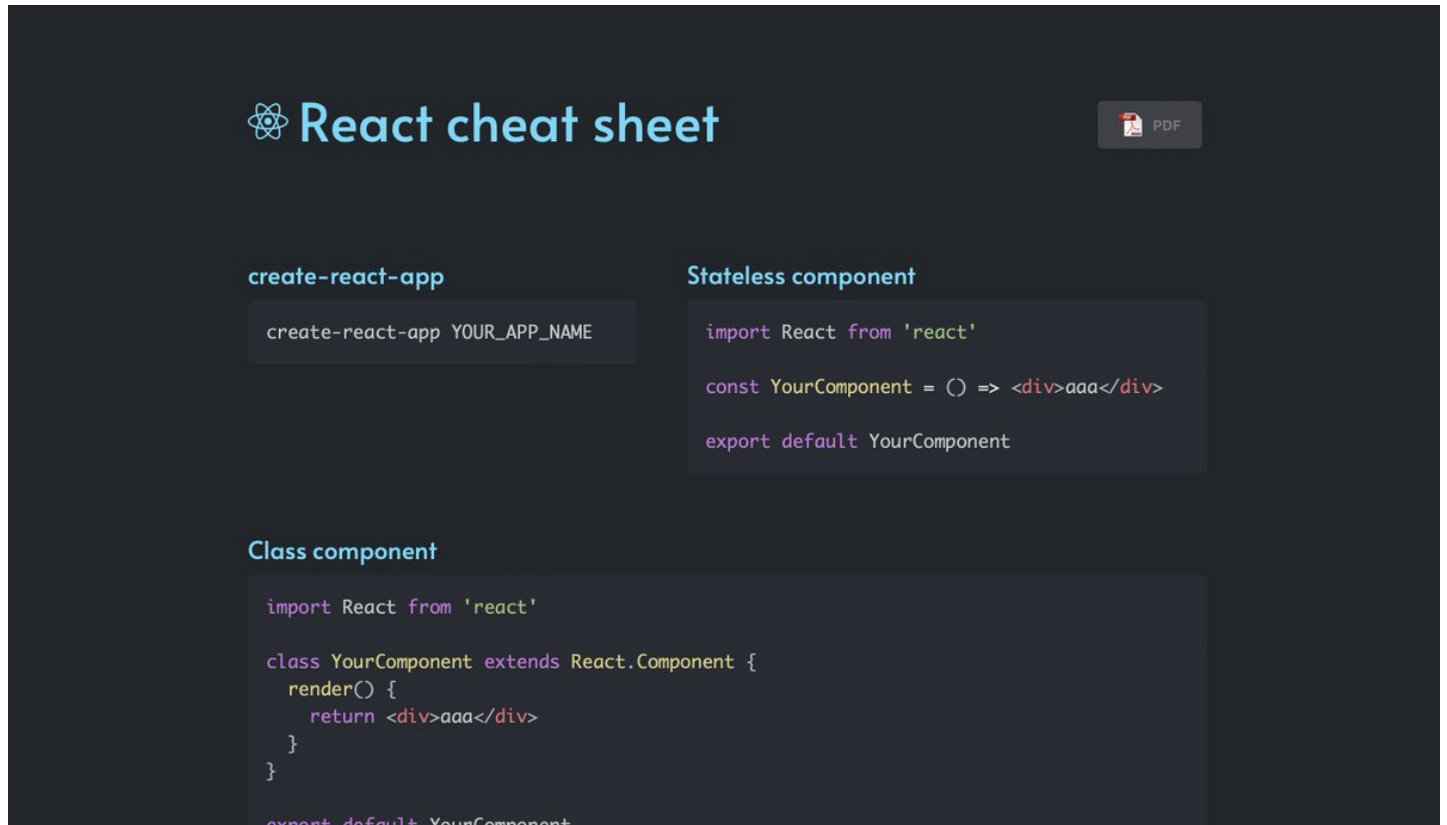
Y

Z

■ Developer Cheat Sheets

This cheat sheet is a compilation of some of the most popular languages: The history of web development, React cheat sheet, React Router cheat sheet, Redux cheat sheet, ES6 cheat sheet, and Request cheat sheet.

■ <https://t.co/tALjaVBCey>



There are literally hundreds of resources out there, and I hope that some of the ones listed here will prove to be useful in your day-to-day work — and most importantly help you avoid some time-consuming, routine tasks.

Happy coding! ■■

That's a wrap! ■■■

If you enjoyed this thread:

1. Follow me [@Insharamin](#) for more of these
2. RT the tweet below to share this thread with your audience <https://t.co/Ef7sL76oMk>

Being developers, we work with many different languages.

Which makes it humanly impossible to remember all its aspects.

This is when a "Cheat Sheet" comes super-handly.

Here's the ultimate list of Cheatsheets for Web Developers\U0001f447\U0001f3fb\u2193

— Insha (@Insharamin) [May 11, 2022](#)