a common computer screen resolution is
1920 pixels in width 1080 pixels in height.
Thanks to responsive design websites can be correctly displayed on any of your devices.
However, today we have many different screens with different resolutions,
which is why responsive design is so important.
And it is complicated by new high resolution screens like the one found on your mobile phone.
These screens group multiple physical pixels into one logical pixel to display smoother images and text.
These are often used in your favorite smartphones to give more high definition visuals like making text images and rounded edges appear smoother and making the individual pixels less visible.
Given all the possibilities and complicating factors,
it would be challenging to develop websites that appear correctly on all kinds of devices if it wasn't for responsive design.
Responsive design is a set of three practices that allows a website
to automatically change its visuals.
In other words, to respond based on the device it is displayed on.
It is the combination of three techniques, flexible grids,
fluid images and media queries.
Firstly, flexible grids are made up of columns, gutters and margins.
The space between the columns is called the gutter and the spaces between the content and the left and right edges of the screen are called margins.
Instead of defining website Element sizes based on pixels, flexible grids are defined in percentage values, allowing them to adjust depending on screen size.
Next you have fluid images by setting the CSS max width property of images to 100%.
The images will scale down smaller if they're containing column becomes narrower than the images size but never grow larger.
This enables an image to scale down to fit in a flexibly sized column rather than overflow it but not grow larger and
become pixelated if the column becomes wider than the image.
Finally, there are media queries that are part of CSS.
They allow developers to query the display size orientation and
aspect ratio to conditionally apply CSS rules.
For example, if you wanted your website background to appear blue on a screen size less than or equal to 700 pixels.
Like on a mobile phone, you could use a media rule to set the background depending on the size of the screen.
Remember that I said responsive design is the combination of flexible grids,
fluid images and media queries.
When these elements are used together, you build a website that will automatically adjust its layout based on the device, thus delivering a responsive grid.
In responsive design,
the pixel value specified is often referred to as the breakpoint.
A breakpoint is the point at which a website's content and
layout will adapt to provide the best possible user experience.
A Breakpoint can function in different ways across three different grids
a fixed grid fluid or for with grids and lastly, hybrid grids.
Let's explore each of these now, firstly,
a fixed grid has fixed with columns and flexible margins.
The fixed grid has a fixed content with that doesn't change in a specific breakpoint range while the flexible margins occupy the remaining space on

screen.
Then we have fluid or full width grids with fluid with columns and
fixed gutters and side margins.
The fluid grid has a flexible content with that goes edge to edge as per
the screen size.
In a fluid grid, columns either grow or shrink to adapt to the available space.
And finally there are hybrid grids that have both fluid width and
fixed with components.
Many responsive design frameworks provide multiple CSS
rules based on different device sizes to provide the best visual experience.
Now you know the importance of responsive design.
In the next few videos, you're going to learn more about bootstrap,
the world's most Popular framework for building responsive mobile 1st Sites.

Bootstrap

Bootstrap is often described as a way to "build fast, responsive sites" and it is a "feature-packed, powerful, and extensible frontend toolkit".

Some people refer to it as a "front-end" framework, and some are trying to be more specific by referring to it as a "CSS framework" or a “CSS library”.

So, what is Bootstrap?

Simply put, Bootstrap is a library of CSS and JavaScript code that you can combine to quickly build visually appealing websites.

Modern web development is all about **components**. Small pieces of reusable code that allow you to build websites quickly. Bootstrap comes with multiple components for very fast construction of multiple components, or parts of components.

Another important aspect of modern development is **responsive grids** which allow web pages to adapt their layout and content depending on the device in which they are viewed. Bootstrap comes with a pre-made set of CSS rules for building a responsive grid.

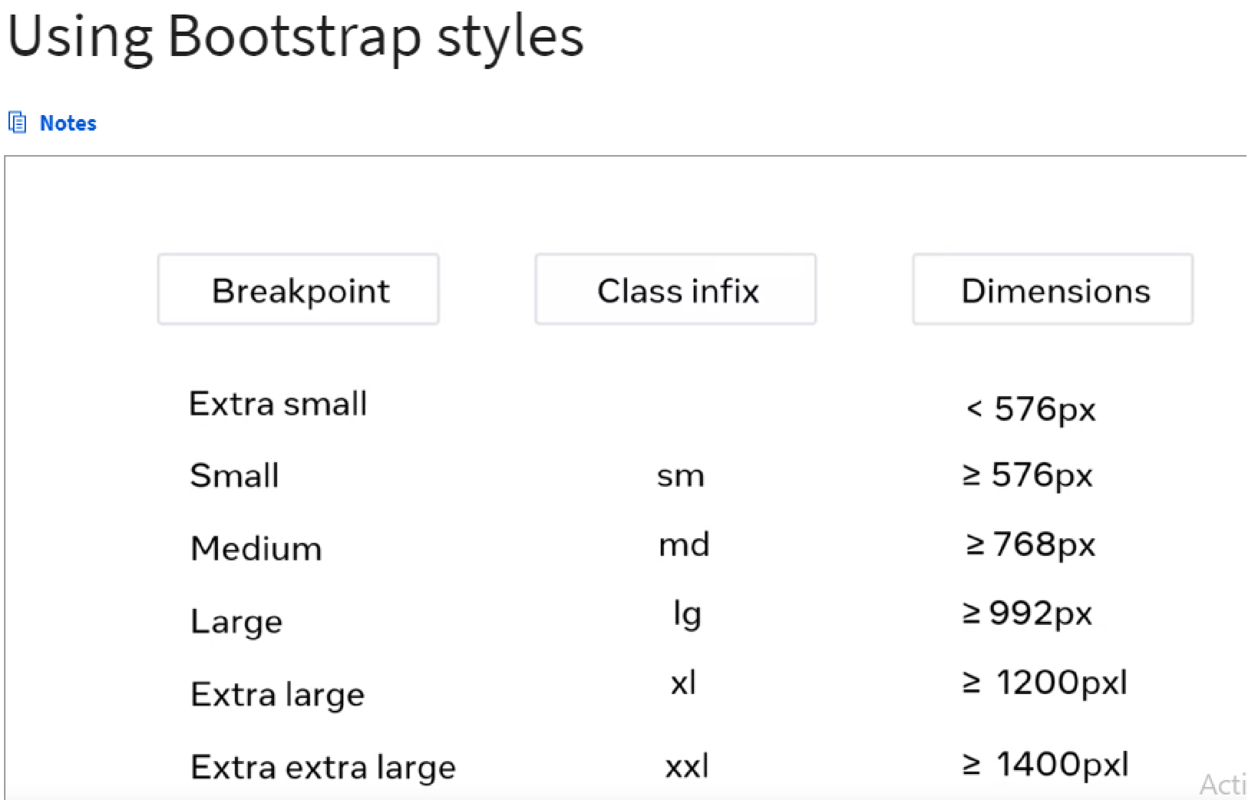
Bootstrap is very popular amongst developers as it saves development time and provides a way for developers to build visually appealing prototypes and websites.

Bootstrap saves significant time because all the CSS code that styles its grid and pre-built components is already written. Instead of having to have a high level of expertise in various CSS concepts, you can just use the existing Bootstrap CSS classes to produce nicely-looking websites. This is indispensable when you need to quickly iterate on website layouts.

Once you know how Bootstrap works, you’ll have enough knowledge to tweak its styling and a whole new world of development opens up to you.

Since Bootstrap is so popular, understanding how to work with it is a prerequisite in many web development companies. Additionally, you can be safe in knowing that both you and your team members have a common design system and you don't have to spend time deciding how to build one. You are free to jump from team to team, from project to project, even from one company to another, and you don't need to re-learn "their way of doing things".

All of these points make investing time to learn Bootstrap a great way to boost your web development skills. In this lesson, you’ll be introduced to the core concepts of Bootstrap and learn how to build web pages using it.

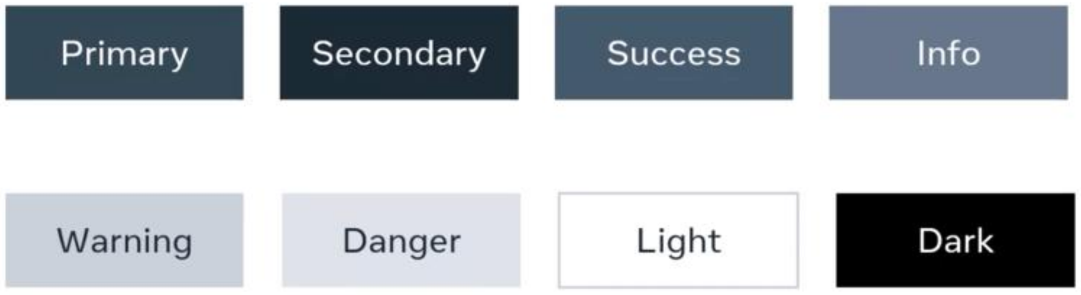


Using Bootstrap styles

Notes

Breakpoint	Class infix	Dimensions
Extra small		< 576px
Small	sm	≥ 576px
Medium	md	≥ 768px
Large	lg	≥ 992px
Extra large	xl	≥ 1200pxl
Extra extra large	xxl	≥ 1400pxl

list of modifiers available in Bootstrap.



React builds a representation of the browser Document Object Model or DOM in memory called the virtual DOM. As components are updated, React checks to see if the component’s HTML code in the virtual DOM matches the browser DOM. If a change is required, the browser DOM is updated. If nothing has changed, then no update is performed.

As you know, this is called the **reconciliation** process and can be broken down into the following steps:

Step 1: The virtual DOM is updated.

Step 2: The virtual DOM is compared to the previous version of the virtual DOM and checks which elements have changed.

Step 3: The changed elements are updated in the browser DOM.

Step 4: The displayed webpage updates to match the browser DOM.

As updating the browser DOM can be a slow operation, this process helps to reduce the number of updates to the browser DOM by only updating when it is necessary.

But even with this process, if a lot of elements are updated by an event, pushing the update to the browser DOM can still be expensive and cause slow performance in the web application.

The React team invested many years of research into solving this problem. The outcome of that research is what’s known as the React Fiber Architecture.

The Fiber Architecture allows React to incrementally render the web page. What this means is that instead of immediately updating the browser DOM with all virtual DOM changes, React can spread the update over time. But what does "over time" mean?

Imagine a really long web page in the web browser. If the user scrolls to the bottom, the top of the web page is no longer visible. The user then clicks a button on the bottom of the web page that updates some text on the top of the web page.

But the top of the page isn't visible. Therefore, why update it immediately?

Perhaps there is text currently displayed on the bottom of the page that also updates when the button is clicked. Wouldn't that be a higher priority to update than the non-visible text?

This is the principle of the React Fiber Architecture. React can optimize when and where updates occur to the browser DOM to significantly improve application performance and responsiveness to user input. Think of it as a priority system. The highest priority changes, the elements visible to the user, are updated first. While lower priority changes, the elements not currently displayed, are updated later.

While you're unlikely to interact with the virtual DOM and Fiber Architecture yourself, it's good to know what's going on if issues occur during the development of your web application.

There are many tools available to help you investigate how React is processing your webpage. The official React Developer Tools web browser plugin developed by Meta will be one of the key tools in your developer toolbox. So, if you do have to look deeper into the code, you'll have the right toolbox available to help you. These tools will be explored later on.

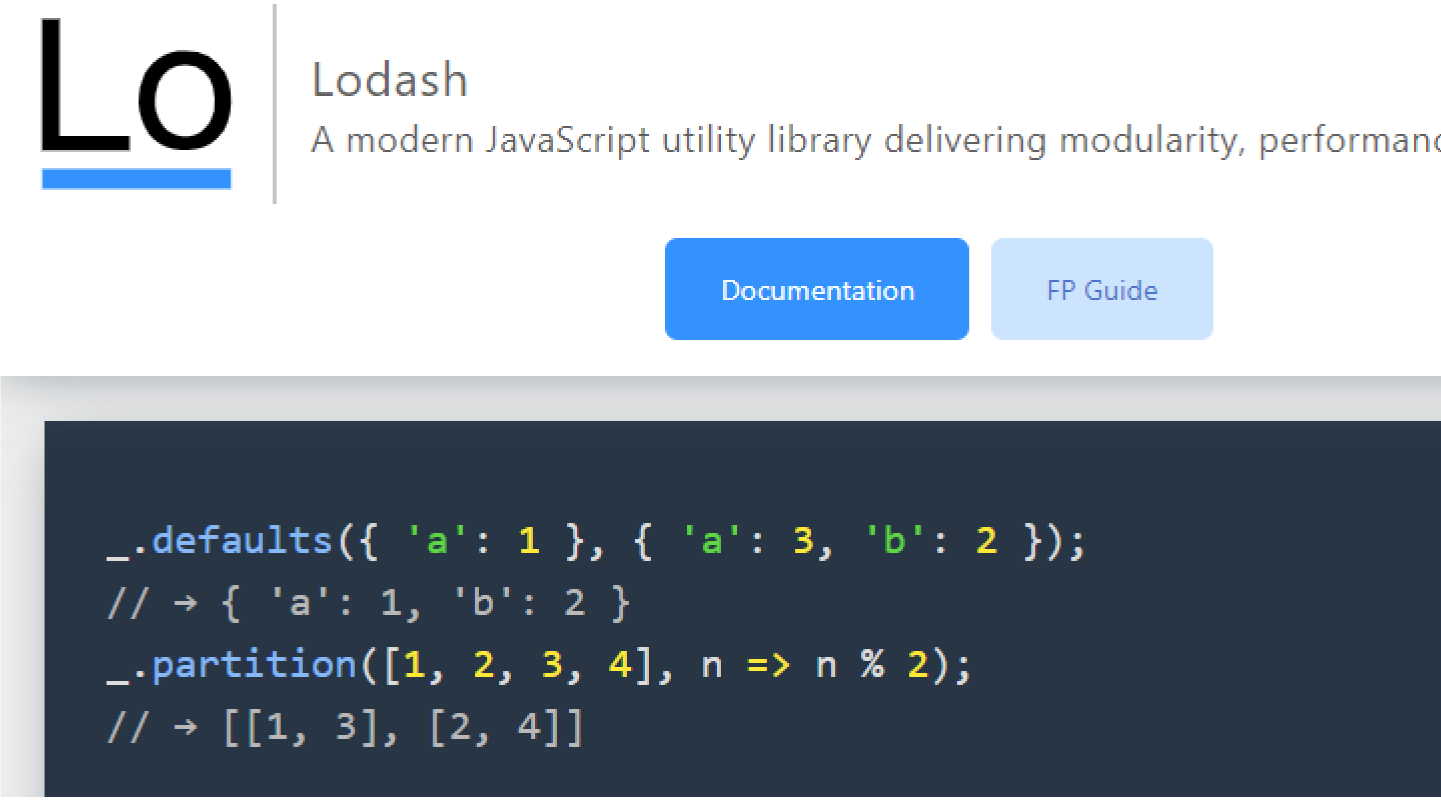
Alternatives to React

React is a library and not a framework. This means you'll often use other JavaScript libraries with it to build your application. In this reading, you will be briefly introduced to some JavaScript libraries commonly used with React.

Lodash

Official Website

As a developer, there's a lot of logic you'll commonly write across applications. For example, you might need to sort a list of items or round a number such as `3.14` to `3`. Lodash provides common logic such as these as a utility library to save you time as a developer.



Lodash

A modern JavaScript utility library delivering modularity, performanc

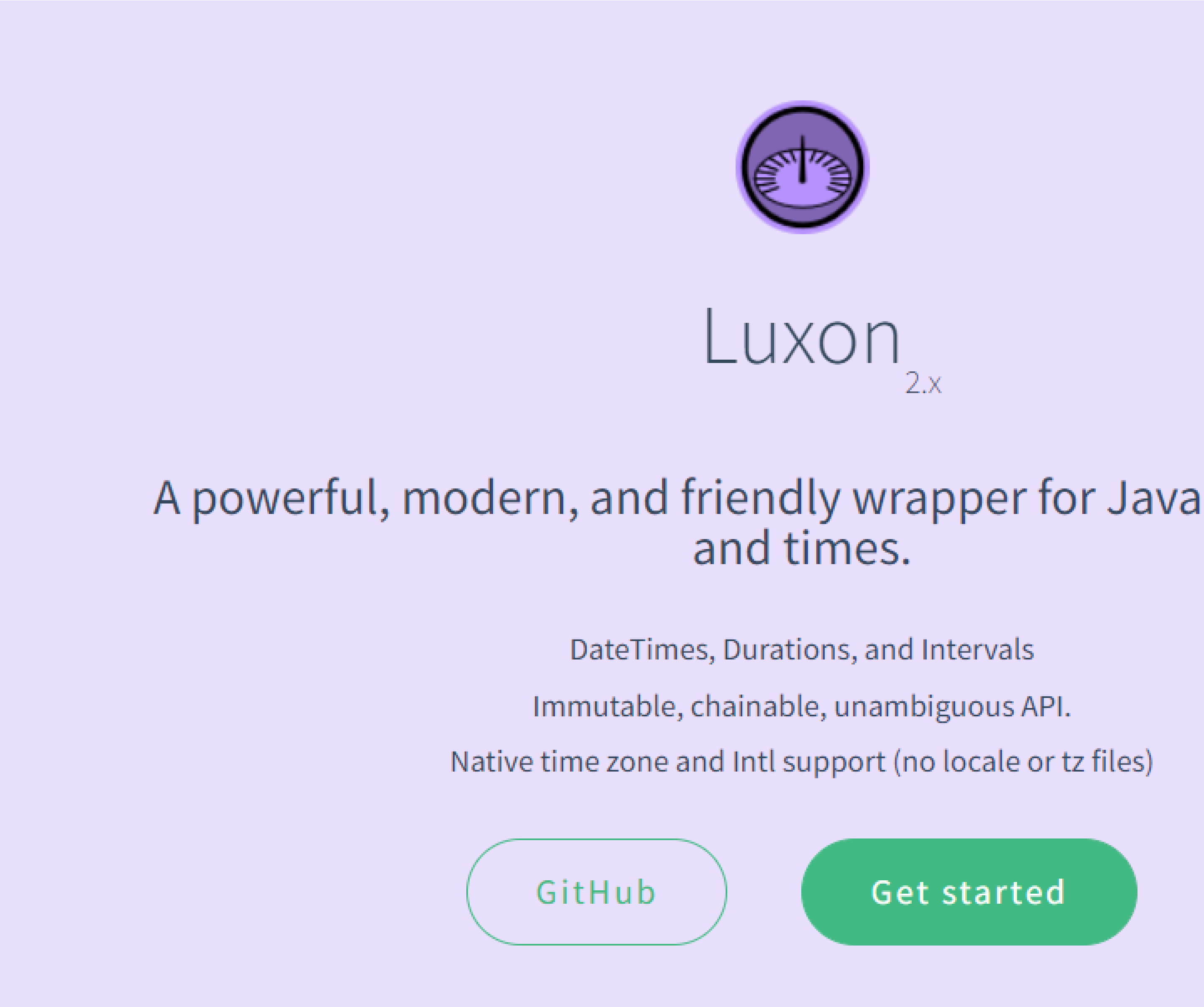
```
_.defaults({ 'a': 1 }, { 'a': 3, 'b': 2 });
// → { 'a': 1, 'b': 2 }
_.partition([1, 2, 3, 4], n => n % 2);
// → [[1, 3], [2, 4]]
```

Luxon

[Official Website]()

You'll be working with dates and times often as a developer. Think of viewing a list of orders and when they were placed, or displaying a calendar schedule for an event. Dates and times are everywhere.

Luxon helps you work with dates and times by providing functions to manipulate and display them. For example, think of how dates are formatted in different countries. In the United States the format is `Month Day Year` but in Europe it is `Day Month Year`. This is one area where Luxon can help you display the date in the user's local format.



Luxon 2.x

A powerful, modern, and friendly wrapper for Java
and times.

DateTimes, Durations, and Intervals

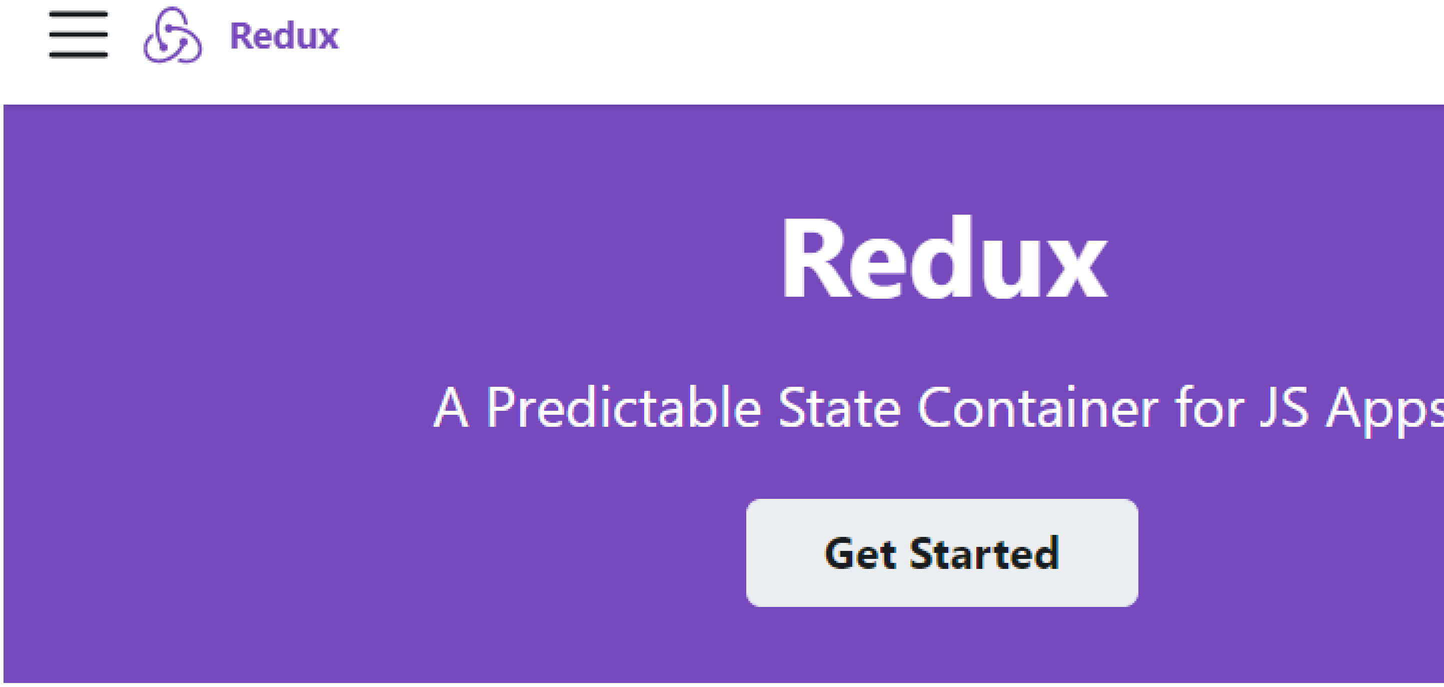
Immutable, chainable, unambiguous API.

Native time zone and Intl support (no locale or tz files)

Redux

[Official Website]()

When building a web application, you'll need to keep track of its state. Think of when you shop online. The web application tracks items currently in your shopping cart. When you remove an item from the cart, the application needs to update what displays on the screen. This is where Redux comes in. It helps you manage your application state and even has advanced features such as undo and redo.



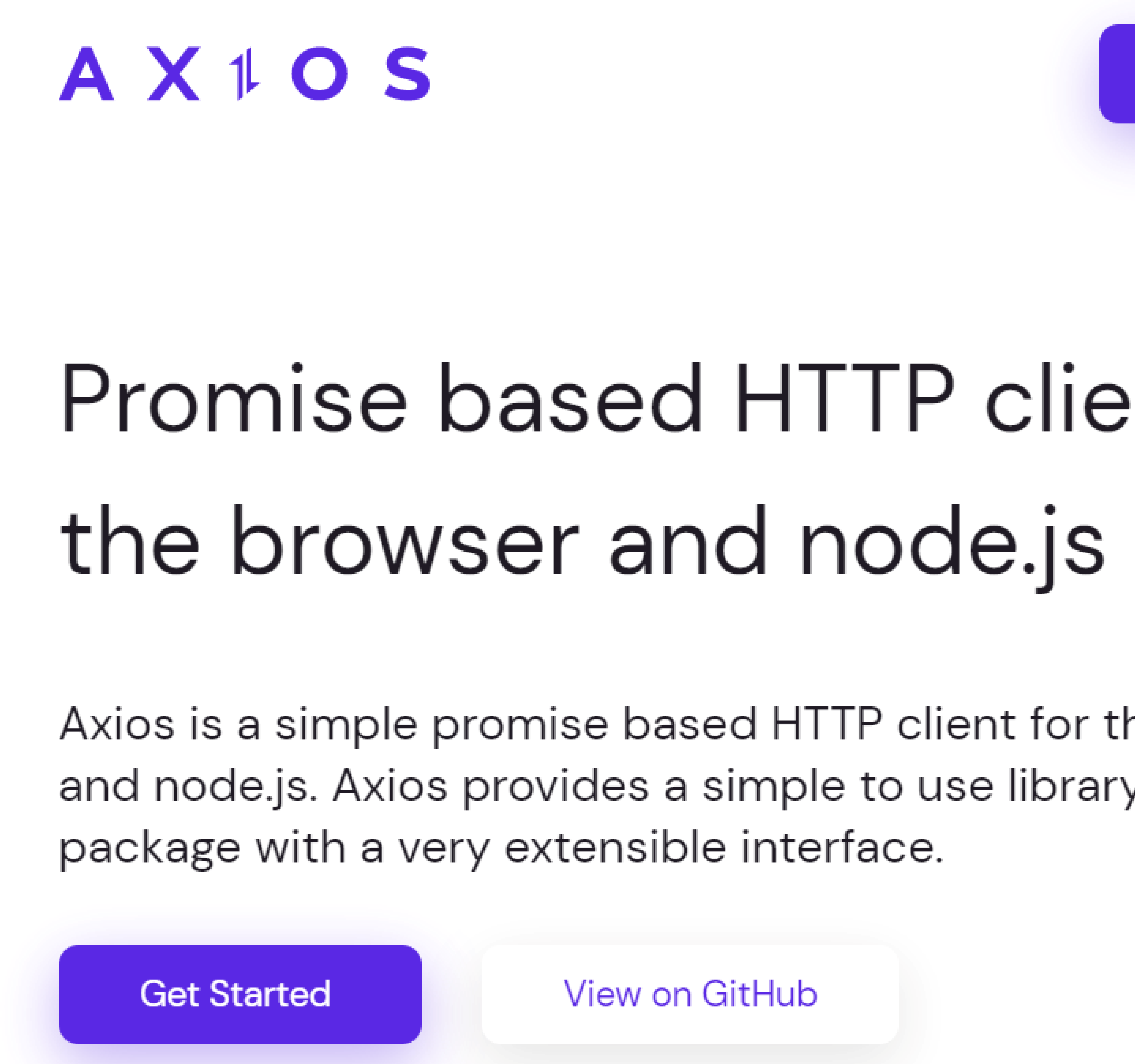
Redux

A Predictable State Container for JS Apps

Axios

Official Website

As a developer you'll be communicating with APIs over HTTP frequently. The Axios library helps to simplify sending HTTP requests and processing the response. It also provides advanced features allowing you to cancel requests and to change data received from the web server before your application uses the data.



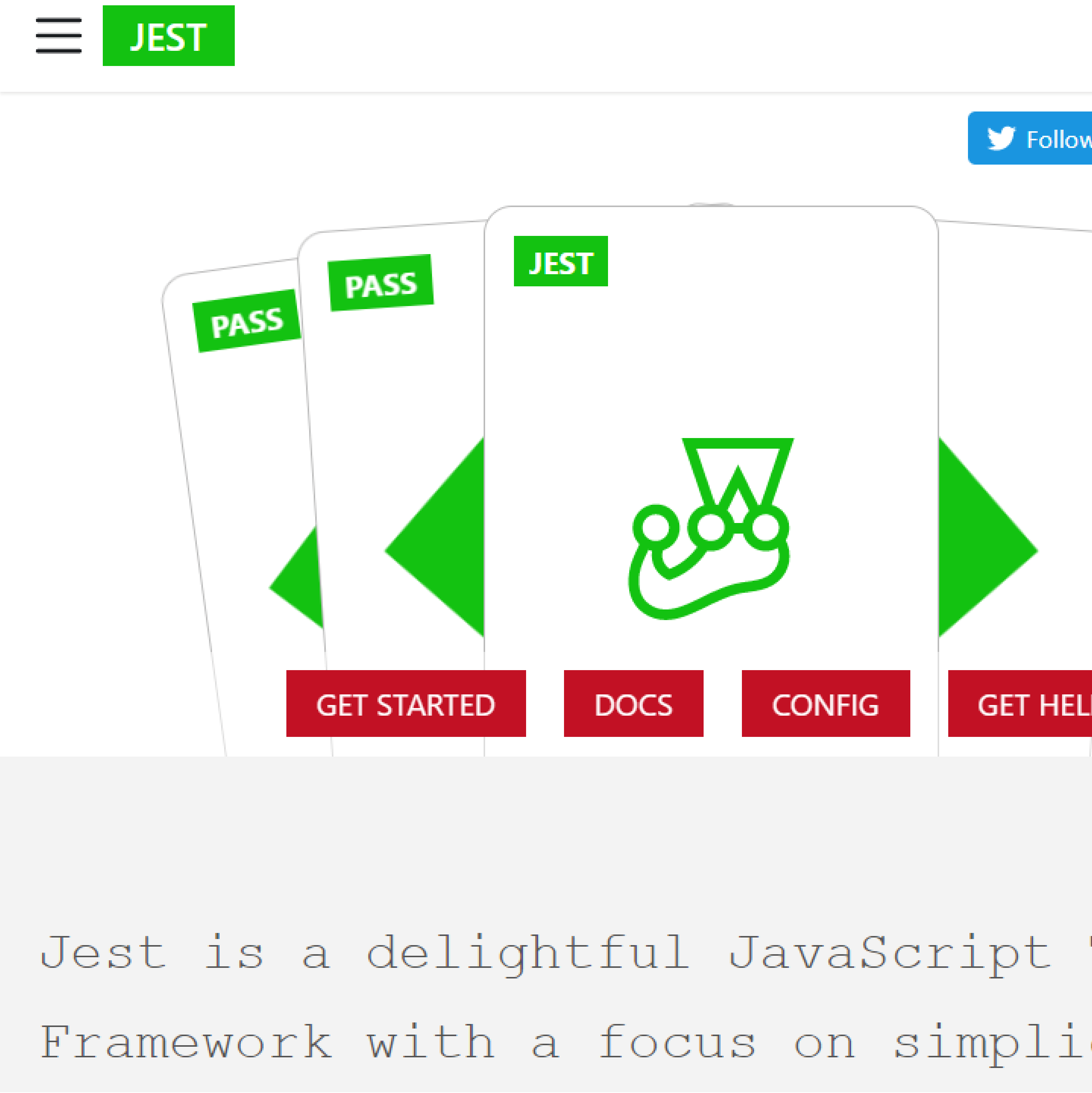
Promise based HTTP clie
the browser and node.js

Axios is a simple promise based HTTP client for th
and node.js. Axios provides a simple to use library
package with a very extensible interface.

Jest

Official Website

It is good practice to write automated tests for your code as a professional developer. The jest library helps you to do this and works with many libraries and frameworks. It also provides reporting utilities such as providing information on how much of your code is tested by your automated tests.



Jest is a delightful JavaScript
Framework with a focus on simpli

Conclusion

If you're curious to learn more about these libraries, their websites feature setup guides, tutorials and documentation to get started. These libraries will be covered later on.

Additional Resources

Learn more Here is a list of resources that may be helpful as you continue your learning journey.

React Official Website

https://reactjs.org/

Choosing between Traditional Web Apps and Single Page Apps (Microsoft)

https://docs.microsoft.com/en-us/dotnet/architecture/modern-web-apps-azure/choose-between-traditional-web-and-single-page-apps

React Source Code (Github)

https://github.com/facebook/react

Introduction to React.js

The original video recorded at Facebook in 2013.

https://youtu.be/XxVg_s8xAms