Database: codecourses, Table: article, Purpose: Dumping data

ID	BODY	INSTRUCTORID	AUTHORFNAME	AUTHORSNAME
1	# Project Title A brief description of what this project does and who it`s for   ## ???? About Me I`m a full stack developer... # Hi, I`m Katherine! ???? ## ???? Links [![portfolio](https://img.shields.io/badge/my_portfolio-000?style=for-the-badge&logo=ko-fi&logoColor=white)](https://katherineoelsner.com/) [![linkedin](https://img.shields.io/badge/linkedin-0A66C2?style=for-the-badge&logo=linkedin&logoColor=white)](https://www.linkedin.com/) [![twitter](https://img.shields.io/badge/twitter-1DA1F2?style=for-the-badge&logo=twitter&logoColor=white)](https://twitter.com/) ## ???? Skills Javascript, HTML, CSS... ## Lessons Learned What did you learn while building this project? What challenges did you face and how did you overcome them? ![Logo](https://dev-to-uploads.s3.amazonaws.com/uploads/articles/th5xamgrr6se0x5ro4g6.png) ## Optimizations What optimizations did you make in your code? E.g. refactors, performance improvements, accessibility ## Roadmap - Additional browser support - Add more integrations ## Used By This project is used by the following companies: - Company 1 - Company 2 ## Screenshots ![App Screenshot](https://via.placeholder.com/468x300?text=App+Screenshot+Here) ## Run Locally Clone the project ```bash git clone https://link-to-project ``` Go to the project directory ```bash	2	admin	aliaa

Database: codecourses, Table: article, Purpose: Dumping data

ID	BODY	INSTRUCTORID	AUTHORFNAME	AUTHORSNAME
	cd my-project ``` Install dependencies ```bash npm install ``` Start the server ```bash npm run start ``` ## Installation Install my-project with npm ```bash npm install my-project cd my-project ```			
4	how to kill react and database before it kill you	2	admin	aliaa
5	how to kill react and database before it kill you	2	admin	aliaa
6	how to kill react and database before it kill you	2	admin	aliaa
7	## how this happen to us when we are so so fine \n \r we get all what we want	2	admin	aliaa
8	## small thing we all new small things in our lide	2	admin	aliaa
9	adhaodifchzpsojdpoAKFS adhaodifchzpsojdpoAKFSadhaodifchzpsojdpoAKFSadhaodifchzpsojdpoAKFSadhaodifchzpsojdpoAKFSadhaodifchzpsojdpoAK FS adhaodifchzpsojdpoAKFS adhaodifchzpsojdpoAKFSadhaodifchzpsojdpoAKFSadhaodifchzpsojdpoAKFS adhaodifchzpsojdpoAKFS adhaodifchzpsojdpoAKFS	2	admin	aliaa
10	akda;dj akda;dj akda;djakda;djakda;djakda;djakda;djakda;djakda;dj	2	admin	aliaa
11	- aliaa - gheis - modern life - comics	2	admin	aliaa
12	- aliaa - gheis - modern life - comics	2	admin	aliaa
13	- aliaa - gheis - modern life - comics	2	admin	aliaa
17	react-howto If you’re new to React (or frontend in general) you may find the ecosystem confusing. There are a few reasons for this. React has historically been targeted at early-adopters and experts Facebook only open-sources what it actually uses, so it doesn’t focus on tooling for smaller-than-Facebook projects There’s a lot of bad marketing masquerading as React guides Throughout this document, I’ll assume you’ve built a web page with HTML, CSS and JavaScript. Why should you listen to me? There’s a ton of conflicting advice about React out there; why listen to me? I was one of the original members of the Facebook team that built and open-sourced React. I’m no longer at Facebook and I’m now at a small startup, so I have a non-Facebook perspective as well.	2	admin	aliaa

ID	BODY	INSTRUCTORID	AUTHORFNAME	AUTHORSNAME
	How to tackle the React ecosystem All software is built on a stack of technologies, and you need to understand enough of that stack to build your app. The reason why the React ecosystem of tooling seems overwhelming is because it’s always explained in the wrong order. You should learn, in this order, without skipping ahead or learning concurrently: React itself npm JavaScript “bundlers” ES6 Routing Flux You don't need to learn all of these to be productive with React. Only move to the next step if you have a problem that needs to be solved. Additionally, there are a few topics that are often mentioned in the React community that are "bleeding edge". The topics below are interesting, but they're difficult to understand, are far less popular than the above topics and aren't required for most apps. Inline styles Server rendering Immutable.js Relay, Falcor, etc Learning React itself It’s a common misconception that you need to waste a lot of time setting up tooling to start to learn React. In the official documentation you’ll find a copy-paste HTML template that you can save in an .html file and get started right away. No tooling is required for this step, and don’t start learning extra tooling until you’re comfortable with React basics. I still think the easiest way to learn React is the official tutorial. Learning npm npm is the Node.js package manager and is the most popular way front-end engineers and designers share JavaScript code. It includes a module system called CommonJS and lets you install command-line tools written in JavaScript. Read this post for background on why CommonJS is necessary for browsers, or the CommonJS Spec Wiki for more on the CommonJS API. Most reusable components, libraries and tools in the React ecosystem are available as CommonJS modules and are installed with npm. Learning JavaScript bundlers For a number of good technical reasons CommonJS modules (i.e. everything in npm) cannot be used natively in the browser. You need a JavaScript “bundler” to “bundle” these modules into .js files that you can include in your web page with a <script> tag. Examples of JavaScript bundlers include webpack and browserify. Both are good choices, but I prefer webpack since it has a lot of features that make development of large apps easier. Since its documentation can be confusing, I have a plug-and-play template for getting started and I wrote a how-to guide for webpack for more complex use cases. React also now offers an officially supported CLI tool called Create React App. It lets you create React projects powered by webpack without any configuration. It has its limitations, but it can serve as a great starting point, and its updates will add more features over time. It also offers an "ejection" feature that copies all configs and dependencies into your project so you have full control over them. One thing to keep in mind: CommonJS uses the require() function to import modules, so a lot of people get confused and think that it has something to do with a project called require.js. For a number of technical reasons, I would suggest that you avoid require.js. It’s also not very popular in the React ecosystem. Learning ES6 Outside of JSX (which you learned in the React tutorial), you may see some funny syntax in React examples. This is called ES6, and it’s the latest version of JavaScript so you may not have learned it yet. Since it’s so new, it’s not supported in browsers yet, but your bundler can translate it for you with the proper configuration. If you just want to get things done with React, you can skip learning ES6, or try to pick it up along the way. You may see some talk about ES6 classes being the preferred way to create React components. This is untrue. Most people (including Facebook) are using React.createClass(). Learning routing “Single-page applications” are all the rage these days. These are web pages that load once, and when the user clicks on a link or a button, JavaScript running on the page updates the address bar, but the web page is not refreshed. Management			

ID	BODY	INSTRUCTORID	AUTHORFNAME	AUTHORSNAME
	of the address bar is done by something called a router. The most popular router in the React ecosystem is react-router. If you’re building a single-page application, use it unless you have a good reason not to. Don’t use a router if you aren’t building a single-page application. Most projects start out as smaller components inside of a larger application anyway. Learning Flux You’ve probably heard of Flux. There’s a ton of misinformation about Flux out there. A lot of people sit down to build an app and want to define their data model, and they think they need to use Flux to do it. This is the wrong way to adopt Flux. Flux should only be added once many components have already been built. React components are arranged in a hierarchy. Most of the time, your data model also follows a hierarchy. In these situations Flux doesn’t buy you much. Sometimes, however, your data model is not hierarchical. When your React components start to receive props that feel extraneous, or you have a small number of components starting to get very complex, then you might want to look into Flux. You’ll know when you need Flux. If you aren’t sure if you need it, you don’t need it. If you have decided to use Flux, the most popular and well-documented Flux library is Redux. There are a lot of alternatives out there, and you’ll be tempted to evaluate lots of them, but my advice is to just stick with the most popular one. Learning inline styles Pre-React, a lot of people reused CSS styles with complicated style sheets built by preprocessors like SASS. Since React makes writing reusable components easy, your stylesheets can be less complicated. Many in the community (including myself) are experimenting with getting rid of stylesheets altogether. This is a fairly crazy idea for a number of reasons. It makes media queries more difficult, and it's possible that there are performance limitations using this technique. When starting out with React, just style things the way you normally would. Once you've got a feel for how React works, you can look at alternate techniques. One popular one is BEM. I recommend phasing out your CSS preprocessor, since React gives you a more powerful way to reuse styles (by reusing components) and your JavaScript bundler can generate more efficient stylesheets for you (I gave a talk about this at OSCON). With that said, React, like any other JavaScript library, will work just fine with a CSS preprocessor. Alternatively, you can also use CSS Modules, more specifically react-css-modules. With CSS Modules you'll still write CSS (or SASS/LESS/Stylus), but you can manage and compose your CSS files like you'd do with inline styles in React. And you don't need to worry about managing your class names using methodologies like BEM, as this will be handled for you under the hood by the module system. Learning server rendering Server rendering is often called "universal" or "isomorphic" JS. It means that you can take your React components and render them to static HTML on the server. This improves initial startup performance because the user does not need to wait for JS to download in order to see the initial UI, and React can re-use the server-rendered HTML so it doesn't need to generate it client-side. You need server rendering if you notice that your initial render is too slow or if you want to improve your search engine ranking. While it's true that Google now indexes client-rendered content, as of January 2016 every time it's been measured it's been shown to negatively affect ranking, potentially because of the performance penalty of client-side rendering. Server rendering still requires a lot of tooling to get right. Since it transparently supports React components written without server rendering in mind, you should build your app first and worry about server rendering later. You won't need to rewrite all of your components to support it. Learning Immutable.js Immutable.js provides a set of data structures that can help to solve certain performance issues when building React apps. It's a great library, and you'll probably use it a lot in your apps moving forward, but it's completely unnecessary until you have an appreciation of the performance implications. Learning Relay, Falcor etc These are technologies that help you reduce the number of AJAX requests. They’re still very cutting-edge, so if you don’t have a problem with too many AJAX requests, you don’t need Relay or Falcor.			
18	“What’s new?” is an interesting and broadening eternal question, but one which, if pursued exclusively, results only in an endless parade of trivia and fashion, the silt of tomorrow. I would like, instead, to be concerned with the question “What is best?,” a question which cuts deeply rather than broadly, a question whose answers tend to move the silt downstream.	1	aliaa	gheis

ID	BODY	INSTRUCTORID	AUTHORFNAME	AUTHORSNAME
	Robert Pirsig When React came out, virtual DOM got everyone talking. It was a breakthrough and, like any good piece of engineering, it was built with carefully considered tradeoffs. The concept was so simple and powerful at the same time that it became the way people introduce and differentiate React from other front-end frameworks and libraries. “React is a view layer that uses virtual DOM for performance.” Another motto you can often hear is “React can be used as the V in MVC.” At the time, downplaying React’s role in application architecture was intentional because React already had too many “seemingly bad” ideas to risk alienating people by adding some more. In fact, React is not at all about virtual DOM. It’s an implementation detail that made React famous, but it overshadowed other concepts that are less shiny but more important in the long run. Now that we’re not surprised by virtual DOM anymore and it is being adopted by other frameworks and libraries, we can focus on examining React’s true strengths: composition, unidirectional data flow, freedom from DSLs, explicit mutation and static mental model. We will examine these topics in next articles. To be continued.			
19	Hmm, let's look at a concrete example to see what the difference is. For example, what if we have a computationally expensive getDataWithinRange() function that returns a filtered dataset, based on a specified dateRange? Because getDataWithinRange() takes some time to run, we'll want to store it in our component's state object and only update it when dateRange changes. CLASS COMPONENT FUNCTION COMPONENT With lifecycle events, we need to deal with all changes in one spot. Our thinking looks something like:When our component loads, and when props change (specifically dateRange), update data In a function component, we need to think about what values stay in-sync. Each update flows more like the statement:Keep data in sync with dateRange class Chart extends Component { state = { data: null, } componentDidMount() { const newData = getDataWithinRange(this.props.dateRange) this.setState({data: newData}) } componentDidUpdate(prevProps) { if (prevProps.dateRange != this.props.dateRange) { const newData = getDataWithinRange(this.props.dateRange) this.setState({data: newData}) } } render() { return (<svg className="Chart" />) } } const Chart = ({ dateRange }) => { const [data, setData] = useState() useEffect(() => { const newData = getDataWithinRange(dateRange) setData(newData) }, [dateRange]) return (<svg className="Chart" />) } Let's pretend that we're either defining getDataWithinRange() outside of the component, or substituting it for inline code that generates the newData. For this simple example, we'll skip that definition so we can focus on the concepts. See how much easier it is to wrap your head around the concept of keeping variables in-sync? CLASS COMPONENT ~~FUNCTION COMPONENT~~	3	aliaa	testsme

ID	BODY	INSTRUCTORID	AUTHORFNAME	AUTHORSNAME
	In fact, this last example was still thinking inside the class-component box. We're storing data in state to prevent re-calculating it every time our component updates. But we no longer need to use state! Here to the rescue is useMemo(), which will only re-calculate data when its dependency array changes.			
20	React/Redux Links Curated tutorial and resource links I`ve collected on React, Redux, ES6, and more, meant to be a collection of high-quality articles and resources for someone who wants to learn about the React-Redux ecosystem, as well as a source for quality information on advanced topics and techniques. Not quite "awesome", but hopefully useful as a starting point I can give to others. Suggestions welcome. Another important resource is the Reactiflux community on Discord, which has chat channels dedicated to discussion of React, Redux, and other related technologies. There`s always a number of people hanging out and answering questions, and it`s a great place to ask questions and learn. The invite link is at https://www.reactiflux.com. You might also want to check out my categorized list of Redux-related addons, libraries, and utilities, at Redux Ecosystem Links. Also see Community Resources for links to other links lists, podcasts, and email newsletters. Finally, I also keep a dev blog at blog.isquaredsoftware.com, where I write about React, Redux, Webpack, and more.	1	aliaa	gheis
21	Last week, Open AI released this new ChatGPT assistant, and it has taken the world by storm. But what is ChatGPT, and how does it work? According to the official announcement: We’ve trained a model called ChatGPT which interacts in a conversational way. The dialogue format makes it possible for ChatGPT to answer followup questions, admit its mistakes, challenge incorrect premises, and reject inappropriate requests. ChatGPT is optimized for dialogue, and this makes it somewhat different from GitHub Copilot. To find out how good it is with Dart & Flutter, I decided to take it for a ride and test it with various coding tasks. So in this article, I'll share some of my experiments with ChatGPT, show you how well it did, and try to give objective answers to the most pressing questions in life: Can you trust ChatGPT to produce correct code for you? □□ How to make the most of it? □□ Will AI take your job away? □□ Ready? Let's dive in! SPONSOR Code with Andrea is free for everyone. Help me keep it that way by checking out this sponsor: 10x your Flutter productivity. 10x your Flutter productivity. Use FlutterFlow's visual builder to generate clean Flutter code and take your Flutter productivity to the next level. FlutterFlow comes with Flutter + Firebase integration, custom code extensibility, web app + multi-language support, and one-click deploy enabling you to ship apps faster than ever. Using ChatGPT with Dart & Flutter: Methodology To get started, I decided to test ChatGPT on various kinds of tasks: Detect and correct common coding errors Explain how code works Write or complete code to solve a given task For code writing tasks, I followed a two-step process: ask to write code to solve a specific, discrete problem verify if the code compiles, runs, and produces the correct output When the output was not as expected, I asked follow-up questions to see if ChatGPT would correct itself. I repeated this process with nine different tasks. You can find all the results below, along with my observations. □□ Note that this is a long article. If you're short on time and want to cut to the chase, feel free to jump to the final results and evaluation. For improved readability, I have included my dialogues with ChatGPT without modifications (rather than pasting screenshots). You can ask the very same questions on the ChatGPT page if you wish - though note that the output can vary, and ChatGPT will produce slightly different responses every time. To learn about how the model was trained, read here and here.	2	admin	aliaa
22	If you've been using Riverpod for some time, you probably know how to declare providers and use them inside your	3	aliaa	testsme

ID	BODY	INSTRUCTORID	AUTHORFNAME	AUTHORSNAME
	widgets. You may also know that providers are global, but their state isn't. But how do providers really work under the hood? Have you ever wondered: When are providers initialized? When and how do they get disposed? What happens when a widget listens to a provider? What is the lifecycle of a provider? How does Riverpod do data caching? This article will answer all these questions and help you: better understand the relationship between providers and widgets learn how data caching works and how it's related to provider lifecycle events choose the most appropriate data caching behaviour according to your needs It will also help you view Riverpod for what it is: a Reactive Caching and Data-binding Framework that helps you solve complex problems (like data caching) with simple code. Data caching is a broad topic, so we'll cover cache invalidation and other advanced techniques in a follow-up article. But for now, we've got plenty to cover already! Ready? Let's go! □□ This article assumes that you already know the basics. If you're new to Riverpod, read this first: Flutter Riverpod 2.0: The Ultimate Guide SPONSOR Code with Andrea is free for everyone. Help me keep it that way by checking out this sponsor: 10x your Flutter productivity. 10x your Flutter productivity. Use FlutterFlow's visual builder to generate clean Flutter code and take your Flutter productivity to the next level. FlutterFlow comes with Flutter + Firebase integration, custom code extensibility, web app + multi-language support, and one-click deploy enabling you to ship apps faster than ever.			
23	Writing Flutter apps got a lot easier with the release of Riverpod 2.0. The new @riverpod syntax lets us use build_runner to generate all the providers on the fly. And the new AsyncNotifier class makes it easier to perform asynchronous initialization with a more ergonomic API, compared to the good old StateNotifier. I've already covered many of the Riverpod 2.0 changes in these two articles: How to Auto-Generate your Providers with Flutter Riverpod Generator How to use Notifier and AsyncNotifier with the new Flutter Riverpod Generator But when it comes to writing tests, things can get tricky, and it can be challenging to get them working. And if we upgrade our code by replacing StateNotifier with AsyncNotifier, we will find that old tests based on StateNotifier will no longer work. So in this article, we'll learn how to write unit tests for AsyncNotifier subclasses. Here is what we will cover: how to work with ProviderContainer and override providers inside our tests how to set up a provider listener using a ProviderSubscription how to verify that the listener is called using the mocktail package Along the way, we'll discover some gotchas and highlight the advantages of testing with listeners vs. streams. By the end of this article, you'll have a better understanding and a clear template for writing unit tests with Riverpod. □□ The official Riverpod docs already include a helpful page about testing, but it doesn't show how to write asynchronous tests for classes with dependencies. This article will fill the gaps.	4	menna	ahmed
24	Writing Flutter apps using Riverpod got a lot easier with the introduction of the riverpod_generator package. Using the new Riverpod syntax, we use the @riverpod annotation and let build_runner generate all the providers on the fly.	3	aliaa	testsme

ID	BODY	INSTRUCTORID	AUTHORFNAME	AUTHORSNAME
	I have already covered all the basics in this article: How to Auto-Generate your Providers with Flutter Riverpod Generator And in this article, we'll take things further and learn about the Notifier and AsyncNotifier classes that were added to Riverpod 2.0. These classes are meant to replace StateNotifier and bring some new benefits: easier to perform complex, asynchronous initialization more ergonomic API: no longer need to pass ref around no longer need to declare the providers manually (if we use Riverpod Generator) By the end, you'll know how to create custom state classes with minimal effort, and quickly generate complex providers using riverpod_generator. Ready? Let's go! □□ This article assumes that you're already familiar with Riverpod. If you're new to Riverpod, read: Flutter Riverpod 2.0: The Ultimate Guide SPONSOR Code with Andrea is free for everyone. Help me keep it that way by checking out this sponsor: 10x your Flutter productivity. 10x your Flutter productivity. Use FlutterFlow's visual builder to generate clean Flutter code and take your Flutter productivity to the next level. FlutterFlow comes with Flutter + Firebase integration, custom code extensibility, web app + multi-language support, and one-click deploy enabling you to ship apps faster than ever.			
25	I’ve already tackled these questions in a previous article, showing how to implement nested navigation with a combination of Stack, Navigator, and Offstage widgets: Flutter Bottom Navigation Bar with Multiple Navigators: A Case Study However, my previous solution had some limitations and caused too many widget rebuilds. And since it was built with the Navigator 1.0 APIs, it didn't support deep linking and navigation by URL. SPONSOR Code with Andrea is free for everyone. Help me keep it that way by checking out this sponsor: 10x your Flutter productivity. 10x your Flutter productivity. Use FlutterFlow's visual builder to generate clean Flutter code and take your Flutter productivity to the next level. FlutterFlow comes with Flutter + Firebase integration, custom code extensibility, web app + multi-language support, and one-click deploy enabling you to ship apps faster than ever. GoRouter vs Beamer The new Router API (also known as Navigator 2.0) was introduced to support deep linking, URL navigation, and additional use cases. In turn, this led to packages such as GoRouter and Beamer, that provide simple yet powerful routing APIs to meet the needs of all Flutter apps across different platforms. So in this article, I'll show you how to implement nested navigation using both GoRouter (by exploring the new ShellRoute API) and Beamer, offering a fair comparison between these two packages for this common use case. Nested navigation is also supported by the AutoRoute package, but we won't cover it here. Refer to the documentation for more details about nested navigation with AutoRoute.	4	menna	ahmed
26	Why is goal setting important when learning English? Nerida says goal setting is important because it reminds you of how far you have come. “There is nothing better than finding your old goals and looking at how challenging and difficult you thought they were. When you see all your progression laid out like that, it can be very motivating,” she says. Asking questions like “What are you trying to improve your English” and “Why are you learning English” is a constant reminder of where you’re going with your language learning journey, according to Maria. She adds that goal setting contributes to higher motivation levels. If language learners think of goals as bricks and learning a language as building a house, Nerida says learners will quickly realise they have to have solid foundations. “That way, we have not got gaps, we can build this beautiful house that's very strong and it can do what you want it to do,” she says.		aliaa	gheis

ID	BODY	INSTRUCTORID	AUTHORFNAME	AUTHORSNAME
	What are examples of good goals? So if goal setting is important in learning English, how do we set good goals? First, context is important. Maria says it’s important to set personal goals within the context of what you’re doing. Ask yourself: Are you preparing for university? Or are you learning for leisure? That will help you set the appropriate learning goals. Another thing to consider is what type of goal you want to set. Again, it depends on your context. Here are some examples of the types of goals you can set for language learning success: Time-based goals An example of a time-based goal is spending a fixed amount of time – say 30 minutes – on something related to English every day or every second day. For example, reading a novel, watching an English drama or listening to an audio book.			
27	Think of technology you have used recently. What problems did you have with it? What solutions did you come up with to solve the problem? First, let’s look at language related to problems. If you are experiencing an issue with a laptop or video conferencing technology for example, you can say: It’s broken down. It needs fixing./It needs to be fixed. It’s out of order. It’s not working. It’s frozen. It has crashed. It doesn’t work. It keeps making this strange noise. We have to sort it out. I can't unmute. I can't hear you./You're on mute. It's not opening. You dropped out there for a second. When talking about problems related to mobile phones, you could use the following: There is no reception. You're breaking up. I can't hear you./You're inaudible. Next, let’s look at language related to solutions. Here is what you can say: Have you tried switching/turning it on and off? Have you tried restarting your device? It's sorted. It’s fixed! It’s working now. It’s all done. I can see you clearly now. If you have tried your best to solve the problem and none of the solutions you had in mind worked, it's alright to say: I'm sorry I don't know what the issue is. Maybe it's best I send you an email. I'm sorry I can't help here. Shall we reschedule to another time?	2	admin	aliaa
28	The other day, my four-year-old boy was asking me, "Mummy, do you know my dinosaurs' names?" I was like, "No baby, what are they?" "They are Triceratops and Stegosaurus," he said clearly. "What? Again, what?" Somehow, my tongue didn't work! "Oh Mummy," he laughed. Later, he turned around and asked my husband, "Daddy, can you say the names? Mummy doesn't know." I was feeling so embarrassed that I couldn't say these dinosaurs' names while my son said them clearly — a four-year-old kid. I have been in Australia since I was a teenager. Although I wasn't born into an English environment, I graduated from an Australian university, got married to someone from a different cultural background, and am in an English-speaking world 24/7. But I still experience some difficulties saying and understanding specific words or slang. I feel this is not as right as my kids can pick up pronunciation and meaning, even if they are only three and four.	4	menna	ahmed

Database: codecourses, Table: article, Purpose: Dumping data

ID	BODY	INSTRUCTORID	AUTHORFNAME	AUTHORSNAME
	"I definitely think children grasp the English language more easily than adults, as adults have a lifetime of their native language," says Michelle Thomas, who has been in the early childhood sector for over 34 years. "Children have a wider capacity to grasp English while retaining their mother language," she adds. "Adults' brains are already wired, and children's brains are still developing."			
29	The APIs available to a Node.js app consist of: The ECMAScript standard library (which is part of the language) Node.js APIs (which are not part of the language proper): Some of the APIs are provided via global variables: Especially cross-platform web APIs such as fetch and CompressionStream fall into this category. But a few Node.js-only APIs are global, too – for example, process. The remaining Node.js APIs are provided via built-in modules – for example, 'node:path' (functions and constants for handling file system paths) and 'node:fs' (functionality related to the file system). The Node.js APIs are partially implemented in JavaScript, partially in C++. The latter is needed to interface with the operating system. Node.js runs JavaScript via an embedded V8 JavaScript engine (the same engine used by Google’s Chrome browser). Global Node.js variables These are a few highlights of Node’s global variables: crypto gives us access to a web-compatible crypto API. console has much overlap with the same global variable in browsers (console.log() etc.). fetch() lets us use the Fetch browser API. process contains an instance of class Process and gives us access to command line arguments, standard input, standard out, and more. structuredClone() is a browser-compatible function for cloning objects. URL is a browser-compatible class for handling URLs. More global variables are mentioned throughout this blog post.	4	menna	ahmed
30	In An Introduction to GraphQL, you learned that GraphQL is an open-source query language and runtime for APIs created to solve issues that are often experienced with traditional REST API systems. A good way to begin understanding how all the pieces of GraphQL fit together is to make a GraphQL API server. Although Apollo GraphQL is a popular commercial GraphQL implementation favored by many large companies, it is not a prerequisite for making your own GraphQL API server. In this tutorial, you will make an Express API server in Node.js that serves up a GraphQL endpoint. You will also build a GraphQL schema based on the GraphQL type system, including operations, such as queries and mutations, and resolver functions to generate responses for any requests. You will also use the GraphiQL integrated development environment (IDE) to explore and debug your schema and query the GraphQL API from a client. Prerequisites To follow this tutorial, you will need: A local Node.js environment, which you can set up by following the How To Install Node.js and Create a Local Development Environment tutorial for your operating system and distribution. An understanding of the fundamental concepts of GraphQL, which you can find in the tutorial, An Introduction to GraphQL. Familiarity with HTTP. A basic knowledge of HTML and JavaScript, which you can gain from the series, How To Build a Website With HTML and How To Code in JavaScript.	4	menna	ahmed
31	AI is currently one of the hottest buzzwords in tech and with good reason. The last few years have seen several innovations and advancements that have previously been solely in the realm of science fiction slowly transform into reality. Experts regard artificial intelligence as a factor of production, which has the potential to introduce new sources of growth and change the way work is done across industries. For instance, this PWC article predicts that AI could potentially contribute $15.7 trillion to the global economy by 2035. China and the United States are primed to benefit the most from the coming AI boom, accounting for nearly 70% of the global impact. This Simplilearn provides an overview of AI, including how it works, its pros and cons, its applications, certifications, and why it’s a good field to master.	2	admin	aliaa
36	The term artificial intelligence was coined in 1956, but AI has become more popular today thanks to increased data volumes, advanced algorithms, and improvements in computing power and storage.	4	menna	ahmed

ID	BODY	INSTRUCTORID	AUTHORFNAME	AUTHORSNAME
	Early AI research in the 1950s explored topics like problem solving and symbolic methods. In the 1960s, the US Department of Defense took interest in this type of work and began training computers to mimic basic human reasoning. For example, the Defense Advanced Research Projects Agency (DARPA) completed street mapping projects in the 1970s. And DARPA produced intelligent personal assistants in 2003, long before Siri, Alexa or Cortana were household names. This early work paved the way for the automation and formal reasoning that we see in computers today, including decision support systems and smart search systems that can be designed to complement and augment human abilities. While Hollywood movies and science fiction novels depict AI as human-like robots that take over the world, the current evolution of AI technologies isn’t that scary – or quite that smart. Instead, AI has evolved to provide many specific benefits in every industry. Keep reading for modern examples of artificial intelligence in health care, retail and more.			
37	How Does Artificial Intelligence Work? What Is AI? Less than a decade after helping the Allied forces win World War II by breaking the Nazi encryption machine Enigma, mathematician Alan Turing changed history a second time with a simple question: “Can machines think?” Turing’s 1950 paper “Computing Machinery and Intelligence” and its subsequent Turing Test established the fundamental goal and vision of AI. At its core, AI is the branch of computer science that aims to answer Turing’s question in the affirmative. It is the endeavor to replicate or simulate human intelligence in machines. The expansive goal of AI has given rise to many questions and debates. So much so that no singular definition of the field is universally accepted. Can machines think? – Alan Turing, 1950 Defining AI The major limitation in defining AI as simply “building machines that are intelligent” is that it doesn't actually explain what AI is and what makes a machine intelligent. AI is an interdisciplinary science with multiple approaches, but advancements in machine learning and deep learning are creating a paradigm shift in virtually every sector of the tech industry. However, various new tests have been proposed recently that have been largely well received, including a 2019 research paper entitled “On the Measure of Intelligence.” In the paper, veteran deep learning researcher and Google engineer François Chollet argues that intelligence is the “rate at which a learner turns its experience and priors into new skills at valuable tasks that involve uncertainty and adaptation.” In other words: The most intelligent systems are able to take just a small amount of experience and go on to guess what would be the outcome in many varied situations.	4	menna	ahmed
38	When one considers the computational costs and the technical data infrastructure running behind artificial intelligence, actually executing on AI is a complex and costly business. Fortunately, there have been massive advancements in computing technology, as indicated by Moore’s Law, which states that the number of transistors on a microchip doubles about every two years while the cost of computers is halved. Although many experts believe that Moore’s Law will likely come to an end sometime in the 2020s, this has had a major impact on modern AI techniques — without it, deep learning would be out of the question, financially speaking. Recent research found that AI innovation has actually outperformed Moore’s Law, doubling every six months or so as opposed to two years. By that logic, the advancements artificial intelligence has made across a variety of industries have been major over the last several years. And the potential for an even greater impact over the next several decades seems all but inevitable.	1	aliaa	gheis
47	his is the first chapter in a step-by-step guide about main React concepts. You can find a list of all its chapters in the navigation sidebar. If you’re reading this from a mobile device, you can access the navigation by pressing the button in the bottom right corner of your screen. Every chapter in this guide builds on the knowledge introduced in earlier chapters. You can learn most of React by reading the “Main Concepts” guide chapters in the order they appear in the sidebar. For example, “Introducing JSX” is the next chapter after this one.	6	Menna	Ahmed
50	Last week, Open AI released this new ChatGPT assistant, and it has taken the world by storm. But what is ChatGPT, and how does it work? According to the official announcement: We’ve trained a model called ChatGPT which interacts in a conversational way. The dialogue format makes it possible for ChatGPT to answer followup questions, admit its mistakes, challenge incorrect premises, and reject inappropriate requests. ChatGPT is optimized for dialogue, and this makes it somewhat different from GitHub Copilot. To find out how good it is with Dart & Flutter, I decided to take it for a ride and test it with various coding tasks.	6	Menna	Ahmed

Database: codecourses, Table: article, Purpose: Dumping data

ID	BODY	INSTRUCTORID	AUTHORFNAME	AUTHORSNAME
	So in this article, I`ll share some of my experiments with ChatGPT, show you how well it did, and try to give objective answers to the most pressing questions in life:  Can you trust ChatGPT to produce correct code for you? ???? How to make the most of it? ???? Will AI take your job away? ???? Ready? Let`s dive in! SPONSOR Code with Andrea is free for everyone. Help me keep it that way by checking out this sponsor: 10x your Flutter productivity. 10x your Flutter productivity. Use FlutterFlow`s visual builder to generate clean Flutter code and take your Flutter productivity to the next level. FlutterFlow comes with Flutter + Firebase integration, custom code extensibility, web app + multi-language support, and one-click deploy enabling you to ship apps faster than ever.			
52	“If you only put into action half of what you read, you’d already be a millionaire.” I’ve heard many variations of this advice, all making the point that the problem isn’t that we don’t have good ideas, but that we implement little of it and therefore our lives don’t change, we don’t lose weight, grow wealth, improve our relationships or become happier. The implication of this advice is that if you read a book and don’t apply it, you’ve just wasted your money. You ought to apply the lessons of what you read, or there was no point reading the book. I disagree.	8	Ali	Sobhy
54	Reading books is cheap. Kindle editions of popular books cost less than a nice meal. Libraries and borrowing books means you often don’t even need to pay that. Even the time spent to read a book is a fairly low investment. For almost any topics there are books which are good enough that reading them isn’t a chore. If you get in the habit you can probably easily read two dozen books a year. Implementing ideas, in contrast, is often quite expensive. Implementing just one idea from a book can take more time, money or effort than reading the book itself. Implementing all the ideas from a single book might take years. The three meanings of meaning in life: Distinguishing coherence, purpose, and significance.	8	Ali	Sobhy
55	An idea from economics you should remember deeply is that when you have diminishing returns from an activity (meaning doing more and more gets less effective), then the optimal amount is when marginal cost equals marginal benefit. To illustrate, imagine a machine that you put in $5 and it spits out money each time. At first it spits out $20 bills. After a while, only $10 bills come out. Eventually only few quarters spit out when you put in your $5. When should you stop using the machine? Obviously, you should stop using the machine when it only gives you $5 back. That’s when marginal benefit (the amount of money you get each time) is equal to marginal cost (the amount you have to put in to run the machine). Now apply this reasoning to books. If a typical book you read costs $20 and requires twenty hours to read, but the value is life changing—you’re not reading enough books! You should keep putting in those $20 and twenty hours until books you read are worth about what you paid for them. Any less and you’re leaving money on the table.	8	Ali	Sobhy
56	Contrary to the popular wisdom, I think a book you never explicitly try to implement can still add value. Perhaps not life-changing, but at the very least, enough value to justify the relatively low cost invested in them. It’s true, most books won’t change your life. But then again, you don’t need a life-changing amount of money and time to consume them. If you pick good books, read them well and think deeply about their implications, that’s enough to earn back their price tag (both in terms of dollars and hours spent). There’s a few ways books have a lot of value, even if you don’t make a direct habit of implementing every idea: 1. Good books limit bad choices. Read enough books about investing and you learn enough to steer away from some clearly bad habits on investing. Yes, the nth book you read may not cause you to change any behavior differently from the n-1 books you read before, but the accumulation of books on personal finance can keep you from spending and saving foolishly. It’s often the things you don’t do after reading a book that justify the investment cost. If a book steers you away from bad strategies which won’t work, that alone can make it worth reading. 2. Reading a lot ensures you have lots of good ideas. The improvements I want to make in my own life and business often resemble a huge (near infinite) list of things I could be doing. I could refine my exercise habits, optimize a landing page, switch to a new productivity app, etc.. The list is usually far larger than I have time to accomplish. That’s okay. What reading a lot of books does is that it increases the overall quality of this list, so that the ideas I’m working on are better. The more you read, the better the average quality of your list, even if you haven’t set aside time to work on any specific idea	8	Ali	Sobhy

Database: codecourses, Table: article, Purpose: Dumping data

ID	BODY	INSTRUCTORID	AUTHORFNAME	AUTHORSNAME
63	Why is goal setting important when learning English? Nerida says goal setting is important because it reminds you of how far you have come. “There is nothing better than finding your old goals and looking at how challenging and difficult you thought they were. When you see all your progression laid out like that, it can be very motivating,” she says. Asking questions like “What are you trying to improve your English” and “Why are you learning English” is a constant reminder of where you’re going with your language learning journey, according to Maria. She adds that goal setting contributes to higher motivation levels. If language learners think of goals as bricks and learning a language as building a house, Nerida says learners will quickly realise they have to have solid foundations. “That way, we have not got gaps, we can build this beautiful house that`s very strong and it can do what you want it to do,” she says. What are examples of good goals? So if goal setting is important in learning English, how do we set good goals? First, context is important. Maria says it’s important to set personal goals within the context of what you’re doing. Ask yourself: Are you preparing for university? Or are you learning for leisure? That will help you set the appropriate learning goals. Another thing to consider is what type of goal you want to set. Again, it depends on your context. Here are some examples of the types of goals you can set for language learning success: Time-based goals An example of a time-based goal is spending a fixed amount of time – say 30 minutes – on something related to English every day or every second day. For example, reading a novel, watching an English drama or listening to an audio book.	6	Menna	Ahmed
64	Think of technology you have used recently. What problems did you have with it? What solutions did you come up with to solve the problem? First, let’s look at language related to problems. If you are experiencing an issue with a laptop or video conferencing technology for example, you can say: It’s broken down. It needs fixing./It needs to be fixed. It’s out of order. It’s not working. It’s frozen. It has crashed. It doesn’t work. It keeps making this strange noise. We have to sort it out. I can`t unmute. I can`t hear you./You`re on mute. It`s not opening. You dropped out there for a second. When talking about problems related to mobile phones, you could use the following: There is no reception. You`re breaking up. I can`t hear you./You`re inaudible. Next, let’s look at language related to solutions. Here is what you can say:  Have you tried switching/turning it on and off? Have you tried restarting your device? It`s sorted. It’s fixed! It’s working now. It’s all done. I can see you clearly now. If you have tried your best to solve the problem and none of the solutions you had in mind worked, it`s alright to say:  I`m sorry I don`t know what the issue is. Maybe it`s best I send you an email. I`m sorry I can`t help here. Shall we reschedule to another time?	6	Menna	Ahmed
65	The w3-container class is the most important of the W3.CSS classes. It provides equality like: Common margins Common paddings Common vertical alignments Common horizontal alignments Common fonts Common colors The w3-container class is typically used with HTML container elements, like: , <header>, <footer>, <article>, <section>, <blockquote>, <form>, and more.	6	Menna	Ahmed
66	W3.CSS Display Classes W3.CSS provides the following display classes: Class Defines w3-display-container Container for w3-display-classes w3-display-topleft Displays content at the top left corner of the w3-display-container w3-display-topright Displays content at the top right corner of the w3-display-container w3-display-bottomleft Displays content at the bottom left corner of the w3-display-container w3-display-bottomright Displays content at the bottom right corner of the w3-display-container w3-display-left Displays content to the left (middle left) of the w3-display-container w3-display-right Displays content to the right (middle right) of the w3-display-container w3-display-middle Displays content in the middle (center) of the w3-display-container w3-display-topmiddle Displays content at the top middle of the w3-display-container	6	Menna	Ahmed

Database: codecourses, Table: article, Purpose: Dumping data

ID	BODY	INSTRUCTORID	AUTHORFNAME	AUTHORSNAME
	w3-display-bottommiddle□Displays content at the bottom middle of the w3-display-container w3-display-position□Displays content at a specified position in the w3-display-container w3-display-hover□Displays content on hover inside the w3-display-container w3-left□Floats an element to the left (float: left) w3-right□Floats an element to the right (float: right) w3-show□Shows an element (display: block) w3-hide□Hides an element (display: none) w3-mobile□Adds mobile-first responsiveness to any element. Displays elements as block elements on mobile devices			

Database: codecourses, Table: article_topic, Purpose: Dumping data

AID	TID
1	1
1	7
1	8
1	14
1	15
1	21
4	4
5	4
6	4
7	4
8	4
9	25
10	1
10	5
10	9
10	16
10	23
17	16
18	16
19	16
20	16
21	18
22	18
23	18
25	18
26	24
27	24
28	24
29	15
30	15
31	25
36	25
37	25
38	25
47	4
47	16
50	17
50	18
52	10
54	10
55	10
56	10
63	24
64	24
65	4
65	7
66	4
66	7

Database: codecourses, Table: choices, Purpose: Dumping data

ID	ISCORRECT	BODY
1	0	no
1	0	yaaas
1	1	yeees
2	0	Fish
2	1	Meat
2	0	Poison
3	0	Cairo
3	1	Helsiniki
3	0	Newzland
3	0	Oslo
4	1	Beijing
4	0	Newzland
4	0	Sharm
5	0	Bejing
5	1	Canberra
5	0	Manofya
5	0	Oslo
6	0	Bejing
6	1	Brasalia
6	0	Oslo
7	1	Flutter is an open-source UI toolkit
7	0	Flutter is an open-source backend development framework
7	0	lutter is an open-source programming language for cross-platform applications
8	0	Facebook
8	1	Google
8	0	Microsoft
9	0	Java
9	1	Dart
9	0	Go
9	0	Kotlin
10	0	four
10	0	six
10	1	two
11	0	Being stressed
11	1	meaningful, satisfying work
11	0	Working only
12	1	existence
12	0	Helping
13	1	Distinguishing coherence
13	1	purpose
13	1	significance.
14	0	Authors can be the friends in your life you wish you had
14	1	Books change the conversation in your head.
14	1	Reading a lot ensures you have lots of good ideas.
15	0	to be lived
15	0	to be living
15	0	to have lived
15	1	to live
16	0	because
16	0	due
16	0	none
16	1	on account of
16	0	owing
17	0	had never said
17	0	have never said
17	1	never said
17	0	not having said
18	1	to be abducted
18	0	to be abducting
18	0	to have been abducted
18	0	to have been abducting
19	1	her
19	0	her own
19	0	hers
19	0	herself
20	1	Displays content at the bottom middle of the w3-display-container
20	0	Displays content at the bottom right corner of the w3-display-container
20	0	Displays content at the top middle of the w3-display-container
21	1	<article>

Database: codecourses, Table: choices, Purpose: Dumping data

ID	ISCORRECT	BODY
21	0	<green>
21	1	<header>
21	0	<hello>

Database: codecourses, Table: course, Purpose: Dumping data

ID	INSTRUCTORID	INSTRUCTORFNAME	INSTRUCTORSNAME	PREREQUISITES
2	2	admin	aliaa	[ ] react [ ] web
15	2	admin	aliaa	- hate react - hate react - hate react
39	6	Menna	Ahmed	There are no formal prerequisites for learning HTML & CSS, but students should be comfortable using computers and navigating files and applications before learning coding skills.
40	6	Menna	Ahmed	There are no formal prerequisites for learning HTML & CSS, but students should be comfortable using computers and navigating files and applications before learning coding skills.
41	6	Menna	Ahmed	Windows setup: Install the Flutter SDK. 7m 19s. Windows setup: Install Android Studio. 2m 39s. Windows setup: Android Emulator. 5m 44s.
42	6	Menna	Ahmed	No prerequistes, Language learning is the process of learning to speak and understand a language, it helps children to acquire practical commands of language.
43	6	Menna	Ahmed	Basics of English
46	1	aliaa	gheis	- hello world - internet - labtop
53	8	Ali	Sobhy	No prerequistes

Database: codecourses, Table: course_topic, Purpose: Dumping data

CID	TID
2	7
2	15
2	21
15	9
15	14
15	15
39	12
39	16
40	7
40	16
41	17
41	18
42	24
43	10
43	24
46	12
46	20
46	21
46	24
46	25
53	9
53	10

Database: codecourses, Table: element, Purpose: Dumping data

ID	CREATIONDATE	TITLE	IMAGE	DESCRIPTION
1	2022-12-22	love quite	https://images.unsplash.com/photo-1672218814630-97eb18814965?ixlib=rb-4.0.3&ixid=MnwxMjA3fDB8MHxwaG90by1wYWdlfHx8fGVufDB8fHx8&auto=format&fit=crop&w=1171&q=80	how to empassion quitness
2	2022-12-22	React Course	https://images.unsplash.com/photo-1672218814630-97eb18814965?ixlib=rb-4.0.3&ixid=MnwxMjA3fDB8MHxwaG90by1wYWdlfHx8fGVufDB8fHx8&auto=format&fit=crop&w=1171&q=80	learn react to hate your life
3	2022-12-22	aliaa gheis in admin	https://images.unsplash.com/photo-1672218814630-97eb18814965?ixlib=rb-4.0.3&ixid=MnwxMjA3fDB8MHxwaG90by1wYWdlfHx8fGVufDB8fHx8&auto=format&fit=crop&w=1171&q=80	How React Killed me ??
4	2022-12-22	welcom home	https://images.unsplash.com/photo-1672218814630-97eb18814965?ixlib=rb-4.0.3&ixid=MnwxMjA3fDB8MHxwaG90by1wYWdlfHx8fGVufDB8fHx8&auto=format&fit=crop&w=1171&q=80	this call from proc this call from pro
5	2022-12-22	welcom home	https://images.unsplash.com/photo-1664574654578-d5a6a4f447bb?ixlib=rb-4.0.3&ixid=MnwxMjA3fDF8MHxwaG90by1wYWdlfHx8fGVufDB8fHx8&auto=format&fit=crop&w=1170&q=80	this call from proc this call from pro
6	2022-12-22	welcom home	https://images.unsplash.com/photo-1672162723391-9fd523f02f69?ixlib=rb-4.0.3&ixid=MnwxMjA3fDB8MHxwaG90by1wYWdlfHx8fGVufDB8fHx8&auto=format&fit=crop&w=1172&q=80	this call from proc this call from pro
7	2022-12-22	hello world	http://localhost:4000/images/5b6ff5fe09448c675eaabe2ef7ad826b	this 's how i greet world in my village
8	2022-12-22	from nodejs from proc	https://images.unsplash.com/photo-1672162723391-9fd523f02f69?ixlib=rb-4.0.3&ixid=MnwxMjA3fDB8MHxwaG90by1wYWdlfHx8fGVufDB8fHx8&auto=format&fit=crop&w=1172&q=80	small summary on miserable how i am
9	2022-12-22	title titlke	https://images.unsplash.com/photo-1664575601711-67110e027b9b?ixlib=rb-4.0.3&ixid=MnwxMjA3fDF8MHxwaG90by1wYWdlfHx8fGVufDB8fHx8&auto=format&fit=crop&w=1171&q=80	coool oasd jsdopjasd
10	2022-12-22	title title	http://localhost:4000/images/385b9f6e67b3b7acf20cadb2b6cac1fc	my life was wasted on react
11	2022-12-22	here we go again	http://localhost:4000/images/4.jpg	test test please test
12	2022-12-22	here we go again	http://localhost:4000/images/4.jpg	test test please test
13	2022-12-22	here we go again	http://localhost:4000/images/4.jpg	test test please test
14	2022-12-22	here we go again	https://images.unsplash.com/photo-1664575601711-67110e027b9b?ixlib=rb-4.0.3&ixid=MnwxMjA3fDF8MHxwaG90by1wYWdlfHx8fGVufDB8fHx8&auto=format&fit=crop&w=1171&q=80	test test please test
15	2022-12-22	hello world by react	https://images.unsplash.com/photo-1604537466573-5e94508fd243?ixlib=rb-4.0.3&ixid=MnwxMjA3fDB8MHxwaG90by1wYWdlfHx8fGVufDB8fHx8&auto=format&fit=crop&w=1169&q=80	my life was wasted on react
16	2022-12-22	hello me	https://images.unsplash.com/photo-1604537466573-5e94508fd243?ixlib=rb-4.0.3&ixid=MnwxMjA3fDB8MHxwaG90by1wYWdlfHx8fGVufDB8fHx8&auto=format&fit=crop&w=1169&q=80	for test trigges in mysql
17	2022-12-28	React-How-to	http://localhost:4000/images/385b9f6e67b3b7acf20cadb2b6cac1fc	Pete Hunt’s guide to the React ecosystem.
18	2022-12-28	You're missing the point of react	http://localhost:4000/images/385b9f6e67b3b7acf20cadb2b6cac1fc	Dan Abramov’s article about the best parts of React.
19	2022-12-28	Thinking in react hooks	http://localhost:4000/images/385b9f6e67b3b7acf20cadb2b6cac1fc	Amelia Wattenberger’s provides visualizations and highlighting the mindset change needed switching from classes to functional components + hooks.
20	2022-12-28	React/Redux links	https://images.unsplash.com/photo-1671975961801-316a3b10df0e?ixlib=rb-4.0.3&ixid=MnwxMjA3fDB8MHxwaG90by1wYWdlfHx8fGVufDB8fHx8&auto=format&fit=crop&w=1332&q=80	Curated tutorial and resource links by Mark Erikson collected on React, Redux, ES6, and more. Very helpful for all kind of developers because of it’s categorised content.
21	2022-12-28	Dart&flutter course1	http://localhost:4000/images/385b9f6e67b3b7acf20cadb2b6cac1fc	Dart & Flutter with ChatGPT: Is it worth it?
22	2022-12-28	Dart&flutter course2	https://images.unsplash.com/photo-1664575601711-67110e027b9b?ixlib=rb-4.0.3&ixid=MnwxMjA3fDF8MHxwaG90by1wYWdlfHx8fGVufDB8fHx8&auto=format&fit=crop&w=1171&q=80	Riverpod Data Caching and Providers Lifecycle: Full Guide
23	2022-12-28	Dart&flutter course3	https://images.unsplash.com/photo-1671975961801-316a3b10df0e?ixlib=rb-4.0.3&ixid=MnwxMjA3fDB8MHxwaG90by1wYWdlfHx8fGVufDB8fHx8&auto=format&fit=crop&w=1332&q=80	How to Unit Test AsyncNotifier Subclasses with Riverpod 2.0 in Flutter
24	2022-12-28	Dart&flutter course4	https://images.unsplash.com/photo-1672162723391-9fd523f02f69?ixlib=rb-4.0.3&ixid=MnwxMjA3fDB8MHxwaG90by1wYWdlfHx8fGVufDB8fHx8&auto=format&fit=crop&w=1172&q=80	How to use Notifier and AsyncNotifier with the new Flutter Riverpod Generator
25	2022-12-28	Dart&flutter course5	https://images.unsplash.com/photo-1604537466573-5e94508fd243?ixlib=rb-4.0.3&ixid=MnwxMjA3fDB8MHxwaG90by1wYWdlfHx8fGVufDB8fHx8&auto=format&fit=crop&w=1169&q=80	Flutter Bottom Navigation Bar with Nested Routes: GoRouter vs Beamer Comparison
26	2022-12-28	English Level1	https://images.unsplash.com/photo-1604537466573-5e94508fd243?ixlib=rb-4.0.3&ixid=MnwxMjA3fDB8MHxwaG90by1wYWdlfHx8fGVufDB8fHx8&auto=format&fit=crop&w=1169&q=80	Setting language learning goals for the new year? Here’s what to do
27	2022-12-28	English Level2	https://images.unsplash.com/photo-1604537466573-5e94508fd243?ixlib=rb-4.0.3&ixid=MnwxMjA3fDB8MHxwaG90by1wYWdlfHx8fGVufDB8fHx8&auto=format&fit=crop&w=1169&q=80	How to talk about tech problems and solutions
28	2022-12-28	English Level3	https://images.unsplash.com/photo-1604537466573-5e94508fd243?ixlib=rb-4.0.3&ixid=MnwxMjA3fDB8MHxwaG90by1wYWdlfHx8fGVufDB8fHx8&auto=format&fit=crop&w=1169&q=80	Parenting a toddler when English is your second language ABC Everyday / By Angie Cui
29	2022-12-28	nodejs lesson1	https://images.unsplash.com/photo-1604537466573-5e94508fd243?ixlib=rb-4.0.3&ixid=MnwxMjA3fDB8MHxwaG90by1wYWdlfHx8fGVufDB8fHx8&auto=format&fit=crop&w=1169&q=80	An overview of Node.js: architecture, APIs, event loop, concurrency

Database: codecourses, Table: element, Purpose: Dumping data

ID	CREATIONDATE	TITLE	IMAGE	DESCRIPTION
			ufDB8fHx8&auto=format&fit=crop&w=1169&q=80	
30	2022-12-28	nodejs lesson2	https://images.unsplash.com/photo-1604537466573-5e94508fd243?ixlib=rb-4.0.3&ixid=MnwxMjA3fDB8MHxwaG90by1wYWdlfHx8fGVufDB8fHx8&auto=format&fit=crop&w=1169&q=80	How To Set Up a GraphQL API Server in Node.js
31	2022-12-28	AI intro	https://images.unsplash.com/photo-1489914099268-1dad649f76bf?ixlib=rb-4.0.3&ixid=MnwxMjA3fDB8MHxwaG90by1wYWdlfHx8fGVufDB8fHx8&auto=format&fit=crop&w=1170&q=80	What is Artificial Intelligence: Types, History, and Future
36	2022-12-28	AI history	https://images.unsplash.com/photo-1604537466573-5e94508fd243?ixlib=rb-4.0.3&ixid=MnwxMjA3fDB8MHxwaG90by1wYWdlfHx8fGVufDB8fHx8&auto=format&fit=crop&w=1169&q=80	Artificial Intelligence History
37	2022-12-28	basics of AI	https://images.unsplash.com/photo-1489914099268-1dad649f76bf?ixlib=rb-4.0.3&ixid=MnwxMjA3fDB8MHxwaG90by1wYWdlfHx8fGVufDB8fHx8&auto=format&fit=crop&w=1170&q=80	ARTIFICIAL INTELLIGENCE DEFINITION: BASICS OF AI
38	2022-12-28	Future of AI	https://images.unsplash.com/photo-1604537466573-5e94508fd243?ixlib=rb-4.0.3&ixid=MnwxMjA3fDB8MHxwaG90by1wYWdlfHx8fGVufDB8fHx8&auto=format&fit=crop&w=1169&q=80	The Future of Artificial Intelligence
39	2022-12-28	HTML Course	http://localhost:4000/images/4.jpg	This course is concerning the basics of html which you need to build up a website.
40	2022-12-28	CSS Course	https://images.unsplash.com/photo-1604537466573-5e94508fd243?ixlib=rb-4.0.3&ixid=MnwxMjA3fDB8MHxwaG90by1wYWdlfHx8fGVufDB8fHx8&auto=format&fit=crop&w=1169&q=80	It`s an intro course in the field of web styling
41	2022-12-28	Flutter Course	http://localhost:4000/images/4.jpg	It`s an intro course in the field of mobile app development
42	2022-12-28	English Course Level1	http://localhost:4000/images/4.jpg	It`s an intro course for learning english.
43	2022-12-28	Advanced English Course	http://localhost:4000/images/4.jpg	It`s a course for whom want to be advance his skills in English.
44	2022-12-28	coool soo cool	http://localhost:4000/images/61c621d169f139fc3e77200fb3c38614	hell kill me twice so i came back
45	2022-12-28	aliaa gheis is awesome	http://localhost:4000/images/ff657387d4be34315a0edf39130bd2a9	How React Killed me ??
46	2022-12-28	cool soo cool me cool	http://localhost:4000/images/4.jpg	my life was wasted on react
47	2022-12-28	hello react	http://localhost:4000/images/4.jpg	Hello React World,dear
48	2022-12-28	Most Liked food	http://localhost:4000/images/4.jpg	What food is most likely?
49	2022-12-28	Capitals Quiz	http://localhost:4000/images/4.jpg	IT`s a quiz about capitals over the world
50	2022-12-28	Flutter Article	http://localhost:4000/images/4.jpg	Dart & Flutter with ChatGPT: Is it worth it?
51	2022-12-28	Flutter Quiz	http://localhost:4000/images/4.jpg	It`s just a quiz about flutter basics
52	2022-12-28	Life Lessons	http://localhost:4000/images/4.jpg	Most Books Won’t Change Your Life (But You Should Read Them Anyways)
53	2022-12-28	How to deal with life	http://localhost:4000/images/4.jpg	This course is prettys imple,it aims to facilitate our lives
54	2022-12-28	Life Lesson2	http://localhost:4000/images/4.jpg	Why You Ought to Read More Advice than You Actually Use
55	2022-12-28	Life Lesson3	http://localhost:4000/images/4.jpg	If the Typical Book Changes Your Life You Read Far Too Few Books
56	2022-12-28	Life Lesson4	http://localhost:4000/images/4.jpg	Why Unused Books Still Have Value
57	2022-12-28	LifeLesson2	http://localhost:4000/images/4.jpg	Quiz on life lesson2
58	2022-12-28	Life Lesson4	http://localhost:4000/images/4.jpg	Quiz on life lesson4
59	2022-12-28	LifeLesson1	http://localhost:4000/images/4.jpg	Quiz on life lesson1
60	2022-12-28	Lesson1	http://localhost:4000/images/4.jpg	Quiz on first Lesson
61	2022-12-28	Lesson2	http://localhost:4000/images/4.jpg	Quiz on second Lesson
62	2022-12-28	Lesson3	http://localhost:4000/images/4.jpg	Quiz on third Lesson
63	2022-12-28	English Lesson1	http://localhost:4000/images/4.jpg	Setting language learning goals for the new year? Here’s what to do
64	2022-12-28	English Lesson2	http://localhost:4000/images/4.jpg	How to talk about tech problems and solutions
65	2022-12-28	CSS lesson1	http://localhost:4000/images/4.jpg	W3.CSS Containers l1
66	2022-12-28	CSS Lesson2	http://localhost:4000/images/4.jpg	W3.CSS Display
67	2022-12-28	CSS Quiz1	http://localhost:4000/images/4.jpg	CSS Quiz1
68	2022-12-28	CSS Quiz2	http://localhost:4000/images/4.jpg	CSS Quiz2

STARTDATE	REVIEWBODY	REVIEWRATING	SID	CID
2022-12-22 05:38:23			1	2
2022-12-22 04:39:53	worst	2	2	2
2022-12-28 21:16:20			6	2
2022-12-28 21:16:35			6	46
2022-12-28 19:17:52			7	40
2022-12-28 19:18:25			7	41
2022-12-28 19:21:37			7	43
2022-12-28 19:22:15			7	46
2022-12-28 19:29:37			8	40
2022-12-28 19:29:53			8	41
2022-12-28 19:30:16			8	42
2022-12-28 19:29:48			8	43
2022-12-28 20:18:53			8	46
2022-12-28 20:39:36			9	2
2022-12-28 20:39:40			9	15
2022-12-28 20:40:04			9	39
2022-12-28 20:39:46			9	40
2022-12-28 20:40:01			9	41
2022-12-28 20:39:52			9	42
2022-12-28 20:39:43			9	43
2022-12-28 20:39:49			9	46
2022-12-28 20:40:16			9	53
2022-12-28 20:53:16			10	2
2022-12-28 20:53:24			10	41
2022-12-28 21:00:03			11	2
2022-12-28 21:00:40			11	40
2022-12-28 21:00:08			11	41
2022-12-28 21:00:14			11	43
2022-12-28 21:00:21			11	46
2022-12-28 21:00:35			11	53

Database: codecourses, Table: instructor, Purpose: Dumping data

ID	RATING
1	2250
2	9700
3	400
4	5300
6	3000
8	2100

Database: codecourses, Table: lesson, Purpose: Dumping data

LID	NAME	DESCRIPTION	CID	QID	AID
11	lesson 1	here is what we do	46	45	18
12	Intro	This is the hello world to flutter	41	51	50
13	Lesson1	First Lesson	53	57	54
15	Lesson2	Second Lesson	53	58	56
16	Lesson1	First Lesson	42	60	63
17	Lesson2	Second Lesson	42	61	64
18	Lesson3	Third Lesson	42	62	64
19	Lesson1	First Lesson	40	67	65
20	Lesson2	Second Lesson	40	68	66

Database: codecourses, Table: likeonarticle, Purpose: Dumping data

UID	AID
1	1
1	4
1	5
1	6
1	8
6	1
6	4
6	6
6	8
6	21
6	27
6	28
6	31
6	37
6	38
7	1
7	4
7	5
7	7
7	9
7	10
7	12
7	28
7	29
7	36
7	37
7	38
7	47
7	50
8	1
8	4
8	30
8	31
8	36
8	37
8	38
8	47
8	50
9	1
9	6
9	21
9	24
9	30
9	36
9	37
9	38
9	47
9	50
9	55
10	1
10	28
10	30
10	31
10	36
10	37
10	38
10	47
10	50
10	52
10	55
10	56
11	1
11	31
11	37
11	38
11	47
11	50
11	55
11	56

Database: codecourses, Table: likeoncomment, Purpose: Dumping data

UID	CID

Database: codecourses, Table: messages, Purpose: Dumping data

MID	SENDDATETIME	TXT	SENDER	CID
1	2022-12-28 19:21:44	hi	Samy6	43
2	2022-12-28 19:21:49	how are you	Samy6	43

Database: codecourses, Table: question, Purpose: Dumping data

ID	SCORE	BODY	INSTRUCTORID
1	50	hello world ???	1
2	4	What is the most likely food?	6
3	2	What is the capital of Finland?	6
4	3	What is the capital of China?	6
5	2	What is the capital of Australia?	6
6	1	What is the capital of Brazil?	6
7	2	What is Flutter?	6
8	2	Who developed the Flutter Framework and continues to maintain it today?	6
9	2	Which programming language is used to build Flutter applications?	6
10	1	How many types of widgets are there in Flutter?	8
11	2	What`s the purpose of life?	8
12	1	What`s the meaning of life?	8
13	2	Choose The three meanings of meaning in life	8
14	2	Why Unused Books Still Have Value	8
15	2	I`m very happy _____ in India. I really miss being there.	6
16	2	They didn`t reach an agreement _____ their differences.	6
17	1	I wish I _____ those words. But now it`s too late.	6
18	2	The woman, who has been missing for 10 days, is believed ____	6
19	3	She was working on her computer with her baby next to _____.	6
20	1	What does this mean w3-display-bottommiddle	6
21	3	w3-container class is typically used with HTML container elements, like:	6

Database: codecourses, Table: quiz, Purpose: Dumping data

ID	MAXSCORE	INSTRUCTORID	INSTRUCTORFNAME	INSTRUCTORSNAME
45	100	1	aliaa	gheis
48	100	6	Menna	Ahmed
49	100	6	Menna	Ahmed
51	100	6	Menna	Ahmed
57	100	8	Ali	Sobhy
58	100	8	Ali	Sobhy
59	100	8	Ali	Sobhy
60	100	6	Menna	Ahmed
61	100	6	Menna	Ahmed
62	100	6	Menna	Ahmed
67	100	6	Menna	Ahmed
68	100	6	Menna	Ahmed

QID	NID	TID
45	1	7
45	1	18
45	1	21
45	1	25
48	2	9
49	3	10
49	3	11
49	4	10
49	4	11
49	5	10
49	5	11
49	6	10
49	6	11
51	7	17
51	7	18
51	8	17
51	8	18
51	9	17
51	9	18
57	11	10
57	12	10
58	13	10
58	14	10
59	11	2
59	11	10
60	15	24
60	16	24
61	16	24
61	18	24
61	19	24
62	15	24
62	16	24
62	17	24
67	20	4
67	20	7
67	21	4
67	21	7
68	20	4
68	20	7

Database: codecourses, Table: quiz_topic, Purpose: Dumping data

QID	TID

Database: codecourses, Table: readarticle, Purpose: Dumping data

SID	AID
1	1
1	4
1	5
1	6
1	7
1	8
1	9
1	10
1	17
1	18
1	20
1	22
1	25
1	26
1	27
1	38
2	1
2	10
3	9
6	1
6	4
6	6
6	8
6	21
6	26
6	27
6	28
6	31
6	37
6	38
7	1
7	4
7	5
7	6
7	7
7	9
7	10
7	11
7	12
7	18
7	28
7	29
7	30
7	36
7	37
7	38
7	47
7	50
8	1
8	4
8	30
8	31
8	36
8	37
8	38
8	47
8	50
8	54
8	55
9	1
9	6
9	21
9	24
9	30
9	36
9	37
9	38
9	47
9	50

Database: codecourses, Table: readarticle, Purpose: Dumping data

SID	AID
9	55
9	56
10	1
10	28
10	30
10	31
10	36
10	37
10	38
10	47
10	50
10	52
10	55
10	56
11	1
11	31
11	37
11	38
11	47
11	50
11	55
11	56

Database: codecourses, Table: student, Purpose: Dumping data

ID	SCORE
1	825
2	100
3	50
5	0
6	550
7	900
8	670
9	626
10	638
11	426

Database: codecourses, Table: studenttakesquiz, Purpose: Dumping data

QID	SID	SCORE	TAKEDATE
45	8	50	2022-12-28
48	8	4	2022-12-28
48	9	4	2022-12-28
48	10	4	2022-12-28
49	10	7	2022-12-28
49	11	8	2022-12-28
51	8	6	2022-12-28
51	10	6	2022-12-28
51	11	2	2022-12-28
57	9	3	2022-12-28
57	11	3	2022-12-28
58	9	4	2022-12-28
59	9	2	2022-12-28
59	10	2	2022-12-28

Database: codecourses, Table: topic, Purpose: Dumping data

ID	NAME
8	all
25	artificial intelligence
15	back
19	bad
23	college
1	cool
7	css
21	database
14	dev
5	django
24	English
18	flutter
16	front
9	good
12	html
10	life
17	mobile
11	modern
6	python
20	sad
22	sections
2	stuff
3	themes
4	web

Database: codecourses, Table: _comment, Purpose: Dumping data

ID	AID	RID	UID	CREATIONDATENTIME	BODY
1	1		2	2022-12-22 15:02:19	hello world
2	1		6	2022-12-28 18:33:07	I think that this article is pretty good
3	4		6	2022-12-28 18:33:31	Great article
4	6		6	2022-12-28 18:33:48	I liked that so much.
5	8		6	2022-12-28 18:34:00	pretty cool
6	27		6	2022-12-28 18:34:14	I liked that too much
7	28		6	2022-12-28 18:34:31	I found that so benefcial
8	21		6	2022-12-28 18:34:49	This article helped me alot
9	38		6	2022-12-28 18:35:24	I am really interseted about this field
10	31		6	2022-12-28 19:06:02	It sounds good
11	1		7	2022-12-28 19:09:37	Pretty good
12	12		7	2022-12-28 19:10:31	Good job
13	7		7	2022-12-28 19:10:54	Ohh ,that`s really great
14	4		7	2022-12-28 19:11:10	I liked it
15	29		7	2022-12-28 19:11:26	that`s fantastic
16	5		7	2022-12-28 19:11:37	Good article
17	6		7	2022-12-28 19:12:02	You are right react is hard
18	9		7	2022-12-28 19:12:25	Never mind.
19	10		7	2022-12-28 19:12:45	Oh what`s the content
20	50		7	2022-12-28 19:13:05	So good
21	47		7	2022-12-28 19:13:30	I am so excited to see more from your articles
22	38		7	2022-12-28 19:13:43	This is so intersting
23	36		7	2022-12-28 19:13:59	Oh, I found that amazing
24	30		7	2022-12-28 19:14:18	I didn`t enjoy it at all.
25	28		7	2022-12-28 19:15:07	That`s very helpful
26	37		7	2022-12-28 19:23:13	That sounds good
27	4	14	8	2022-12-28 19:25:20	I liked it too
28	4		8	2022-12-28 19:25:29	Amazing
29	50		8	2022-12-28 19:25:55	That`s very helpful
30	38		8	2022-12-28 19:26:13	Oh, I found that amazing
31	38		8	2022-12-28 19:26:56	Oh, I found that amazing
32	47		8	2022-12-28 19:37:10	I liked that so much.
33	37		8	2022-12-28 19:37:46	That`s very good
34	31		8	2022-12-28 19:38:14	Great article
35	30		8	2022-12-28 19:38:38	I liked that so much.
36	36		8	2022-12-28 19:39:24	Amazing
37	1		9	2022-12-28 20:37:01	Good job
38	38		9	2022-12-28 20:37:14	Doing Great
39	37		9	2022-12-28 20:37:33	I am interseted
40	55		9	2022-12-28 20:37:48	I liked that so much
41	47		9	2022-12-28 20:37:59	Prettu good
42	50		9	2022-12-28 20:38:19	That`s great
43	36		9	2022-12-28 20:38:36	Ohh, great
44	36		9	2022-12-28 20:38:38	Ohh, grea
45	30		9	2022-12-28 20:39:07	Oh, I found that amazing
46	21		9	2022-12-28 20:39:18	Amazing
47	6		9	2022-12-28 20:42:16	Great article
48	24		9	2022-12-28 20:42:32	That sounds great
49	47		10	2022-12-28 20:48:38	Great article
50	50		10	2022-12-28 20:48:49	I liked that so much.
51	38		10	2022-12-28 20:49:10	I liked that so much.
52	37		10	2022-12-28 20:49:27	Ohh, Good
53	55		10	2022-12-28 20:49:53	I loved it
54	31		10	2022-12-28 20:50:06	I loved it
55	36		10	2022-12-28 20:50:23	that`s fantastic
56	52		10	2022-12-28 20:50:41	that helped me alot
57	30		10	2022-12-28 20:51:01	Great article
58	28		10	2022-12-28 20:51:38	I love dit
59	28		10	2022-12-28 20:51:41	I loved it
60	1		11	2022-12-28 20:57:51	I loved it
61	37		11	2022-12-28 20:58:03	Wow that`s good
62	38		11	2022-12-28 20:58:19	Amazing
63	50		11	2022-12-28 20:58:34	that helped me alot
64	56		11	2022-12-28 20:58:53	Life is better with good people
65	47		11	2022-12-28 20:59:20	React is pretty simple
66	55		11	2022-12-28 20:59:34	Great article
67	31		11	2022-12-28 20:59:57	that`s fantastic

Database: codecourses, Table: _user, Purpose: Dumping data

ID	USERNAME	FNAME	SNAME	EMAIL	JOINDATE	ABOUT	ISADMIN	_PASSWORD	_IMAGE
1	aliaagheis	aliaa	gheis	aliaagheis@gmail.com	2022-12-22	passionate front web developer	0	$2a$10$e552Q1pn.kDGz2jy6/lRtezvv.cJzmlpkzChW..8WA5QszyTwVqbW	http://localhost:4000/images/2f75a1b7184928f87693872c
2	admin	admin	aliaa	admin@admin.com	2022-12-22	so adminic	1	$2a$10$ok/G50rBVkY7ky0KdBRA1eU5iNtoqr7Yz9UmOTQRjLp1lZYzKBvai	http://localhost:4000/images/3e4f15f4c2e60aae5fa07e27
3	aliaagheis1	aliaa	testsme	aliaagheis1@gmail.com	2022-12-22		0	$2a$10$MZFLsL99vOaAzoBguK0c0uQBLYYJJnz9oKYlku53aDhraU3vzYqJe	https://7wdata.be/wp-content/uploads/2016/05/icon-user-
4	menna17	menna	ahmed	menna@gmail.com	2022-12-28		1	0000000000000001111	
5	menna99	Menna	Ahmed	mennaahmed0701@gmail.com	2022-12-28		0	$2a$10$X3ESvgWpXwALhhbvqhYwJ.OmOknkE/ionPxnUr3GNj6pCvFFBgmge	https://7wdata.be/wp-content/uploads/2016/05/icon-user-
6	menna99999	Menna	Ahmed	mennaahmed001@gmail.com	2022-12-28		0	$2a$10$6CrpY/qQzYWjExNihvlLR.zZne1SpXoaWaIwNXkicBH4js8KoObe2	https://7wdata.be/wp-content/uploads/2016/05/icon-user-
7	Samy6	Samy	Ahmed	Samy6@gmail.com	2022-12-28		0	$2a$10$SmYXR/WA7H8TPeSj5LpGeuitgTN9u4CeS1JlOsVfTh1V0Quyzfx02	https://7wdata.be/wp-content/uploads/2016/05/icon-user-
8	Ali9	Ali	Sobhy	Ali9@gmail.com	2022-12-28		0	$2a$10$aBmlCOA8Enr9qursJbY2jenelVhMHooAPlbmGnauLoJuw3RpMSL2G	https://7wdata.be/wp-content/uploads/2016/05/icon-user-
9	Fatma7	Fatma	Ali	Fatma7@gmail.com	2022-12-28		0	$2a$10$PquETuXV454y1jGdCWGEaOxNBrdb6TGqEKfXHov11wW2/Wq3e1JXK	https://7wdata.be/wp-content/uploads/2016/05/icon-user-
10	Mazen9	Mazen	Mohsen	Mazen9@gmail.com	2022-12-28		0	$2a$10$qeTJG93bcerg..Lo2POu0OglRpE1nM6BDQA6U1uGbajRXukat7YUm	https://7wdata.be/wp-content/uploads/2016/05/icon-user-
11	Fady9	Fady	Karem	Fady9@gmail.com	2022-12-28		0	$2a$10$q9MPMFBRgUua7w34gD.9VuGiHjIeTFia15igzJSF/yRsVtEuAWyVi	https://7wdata.be/wp-content/uploads/2016/05/icon-user-