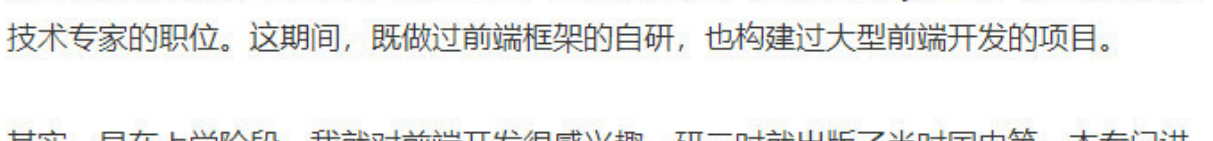


开篇词 | 全面拥抱 Hooks，掌握最新 React 开发方式

王沛 2021-05-24



你好，我是王沛。欢迎和我一起學習 React。

我在前端領域深耕 15 年了，先後工作於 IBM 和 SAP，目前在 eBay 擔任中國區資深研發技術專家的職位。這期間，既做過前端框架的自研，也構建過大型前端開發的項目。

其實，早在上學階段，我就對前端開發很感興趣，研二時就出版了當時國內第一本專門講 Web 2.0 的書籍：《征服 Web 2.0: Ajax 開發技術詳解》。之後的工作也一直專注於前端框架的開發。可以說，我經歷了前端的整個發展歷史，也深知前端開發的各種痛點、難點，並見證了它們是如何被一個個出現的框架和工具不斷攻破的。

而在前端不斷推陳出新的技術中，React 的出現讓我眼前一亮，它解決了很多前端開發的本質性問題，如數據流、狀態維護等，於是我也開始將 React 應用在自己的項目中。經過數十個大型項目的實操，我把自己對 React 的一些可重用的經驗，做成了開源項目，比如 Rekit、React Form Builder 等，在 GitHub 上有超過 5000 個 Star。

同時，作為國內最早一批的 React 實踐者，我也不斷將自己使用 React 的經驗，通過 InfoQ 專欄、QCon 演講和諸多同行分享，推動了 React 在國內的普及；每年也都會在公司內部開設培訓課程，幫助各個團隊迅速掌握 React 的常見設計模式和最佳實踐。

隨著對 React 的深入研究、實戰操作、案例積累，以及輸出分享，我擁有了一套從核心原理到實戰的 React Hooks 知識總結。所以這一次，我想把我多年積累的經驗分享給你。

React，前端開發的正確打開方式

和其他技術一樣，前端開發技術在不斷地更迭變幻。正因如此，面對某個新技術時，我一直抱著非常謹慎的態度。

但是 React 的出現，讓我感覺這才是前端開發的正確打開方式。這主要是因為**在當時它有兩個非常顛覆式的創新點**。

第一個是**推出了虛擬 DOM 機制**，大幅度降低了 UI 上的操作難度。

React 會自動用最優的方式將虛擬 DOM 更新到真實的 DOM 節點，這樣一來，我們就不需要再使用瀏覽器 API 來維護複雜的 DOM 結構了，這就簡化了我們在 UI 上的操作。

第二個是**JSX 語法的使用**，不僅使代碼前所未有的簡潔和直觀，同時也減輕了內部溝通成本，提高了維護效率。

React 用類似 HTML 標記的方式去實現組件化，數據變化以後，UI 也會自動變化，我們再也不用去關注實現細節了，而且保證了 UI 的一致性。

這兩個機制完美地解決了當時前端開發的痛點：模板語言不夠靈活，DOM API 又太繁瑣，且難以維護。因此，React 在 2013 年首次宣布以後，就迅速獲得了大家的認可，成為了最為流行的前端框架。

從下面這張圖也可以看到，React 自誕生以來，下載量就處於一直上升的趨勢。這在前端輪子倍出的背景之下，可以說是一個不俗的成績了。



值得注意的是，在其他前端框架發生翻天覆地的變化的同時，React 的 API 則非常穩定。所以在我看來，無論你是**專職從事前端開發，還是以後端為主，但也需要自己做一些 UI 界面，React 作為目前最為流行的前端框架，基本上都是你的首要選擇**。

但即便在這樣穩定的背景下，React 還是在 2019 年 2 月發布的 16.8 版本中引入了 Hooks 概念。在讓人耳目一新的同時，也让广大早就恐懼於各種新輪子的前端同學心頭一緊，產生本能的抗拒：

- Class 組件不是挺好的嗎，為什麼又要學一個新輪子？
- Hooks 剛出來一兩年，會不會曇花一現？
- 現在項目代碼都是 Class 的，難道都得改一遍嗎？

我先給你吃一顆定心丸，這種擔憂完全沒有必要，這不僅不會增加你的使用難度，反而還會**拓展你的 React 開發思路，為你提供了一種更多的選擇**。

全面擁抱 React Hooks，讓前端代碼結構更加清晰

事實上，在 Hooks 剛出來時，我帶領的團隊中也有很多同學對 Hooks 產生了抗拒心理。這很正常，畢竟，我們初次使用任何新技術時大都會有懷疑心理，或者是畏懼。正如你第一次走一條不熟悉的路時，總會覺得前路漫漫，有點兒迷茫。

即便如此，出於對 React 官方團隊的信賴，我們還是決定開始嘗試。驚喜之處就在於，在我們不斷了解和使用之後，逐漸發現 Hooks 真是太好用了。原來很複雜的一些問題，現在用 Hooks 就可以很優雅地解決！

举个例子来说。在 Hooks 出現之前，**業務邏輯的重用**可以說是 React 開發的一大痛點和難點。比如，你需要在組件中去監聽窗口大小的變化，以便在布局上做調整。那麼我們就得分類組件的不同生命週期中做事件監聽的綁定和解綁。

其中的困難和複雜可想而知，畢竟，這種需要分散在不同方法中的邏輯是很難去重用的。因此，我們不得不借助高階組件這樣複雜而又晦澀的設計模式來解決。

但現在呢，有了 Hooks，我們就能很輕鬆地應對了。我們用 Hooks 就可以非常自然地去解決。我們只需要通過 Hooks 的方式進行封裝，將其變成一個可綁定的數據源。這樣當窗口大小發生變化時，Hook 的組件會自動重新渲染。同時，代碼也更加易懂，並且不會產生額外的組件節點。

可以說，引入 Hooks 的概念之後，函數組件就具備了狀態管理、生命週期管理等能力，**幾乎可以实现原来的 Class 組件具有的所有能力**。函數組件能和已有 Class 組件無縫銜接，一起工作。這樣老的代碼無需做任何改變，新的代碼可以根據具體情況採用 Hooks 的方式去實現。這些都大大加速了 Hooks 的普及。

而且，函數組件結合 Hooks 帶來的**簡潔的寫法，以及直觀的邏輯重用能力**，讓函數組件成為了 React 中的一等公民。**目前，在整个 React 生态中，几乎所有开发项目都已经是围绕着 Hooks 展开的了。**

所以，在这次课程中，我会**从函数组件和 Hooks 的角度出发，带你重新认识 React**，帮助你在深入理解 Hooks 的基础上，使用 Hooks 的思想去开发前端应用，真正享受 Hooks 带来的诸多好处。

不过，对于初次接触 Hooks 的同学来说，肯定会经常遇到各种问题。比如：

- 為什麼編譯又不過了？
- 我的函數組件怎麼又死循環了？
- Redux 的 Store 要怎麼訪問？

有這些問題是很正常的，因為**從 Class 組件到函數組件的轉變，不僅僅是一個寫法的區別，更是整個開發思路的轉變。而我們只有學會用 Hooks 的角度去思考問題，才算把這條路真正走順**。

那麼，在這個課程裡面，我會將我在使用 Hooks 的過程中，以及作為面試官在面試的過程中遇到的、看到的各種痛點問題，以及最需要理解的核心原理呈現出來，幫助你把路走順，且少走彎路，快速掌握並使用 React Hooks 思想進行前端開發。

想用好 React，你該這麼學

現在，你一定覺得 Hooks 簡直太神奇了，迫不及待地想要學習。別急，深入技術點之前，你得知科學的學習路徑。

對於 React Hooks 而言，在我看來最重要的是**準確地知道 Hooks 的功能邊界，也就是知道它能做什麼**。否則，就有可能陷入什麼功能都想用 Hooks 去做的，或者是在最適合使用 Hooks 的功能中，卻想不到用它去處理。

在建立起對 Hook 的全局認知之後，就要掌握**核心概念和原理**了。任何技術，脫離了原理本身，學到的大概率只是皮毛。原理或許不能讓你即學即用，但這却是你**應對實際複雜業務問題的關鍵能力**。

當然，**把原理和實際業務場景結合起來**，才能讓自己在 Hooks 进阶这条路上走得更快、更稳。所以，你就必须清楚日常开发中的具体场景，在真实的场景中透彻理解 Hooks 是如何解决问题的。而我们最高阶的目标，就是让你在**实战中掌握 Hooks 的思考方式，拥有举一反三的能力**。

刚刚说的这些，其实也正是咱们这个课程的核心设计理念。据此，我把学习这门课的路径画成了一张图，你可以对照着来学习。



课程是如何设计的？

接下来，我就说说这门课具体是怎么设计的。整个课程会分为三个部分。

基础篇：抓住问题本质，提高解决复杂问题的能力

刚才我们讲到，原理的学习是必要且重要的，那么，为了让你快速消化吸收这些原理，学得有趣，我也做了一些精心的小设计。

在“基础篇”，我会抽丝剥茧，从“问题的本质是什么”这一角度出发来讲解每个机制，包括：React 为什么要发明 Hooks？useEffect 就是生命周期函数吗？等等。理解了最核心的那个问题，你也就拥有了举一反三的能力。

实战篇：业务场景驱动，掌握解决实际问题的思考方式

在理解核心概念的基础之上，我们就拥有了解决问题的抓手，接下来，我会带你进入实战部分。

在这里，我精心设计了 9 个不同的典型业务场景，并基于 React Hooks 去剖析一些常见设计模式。相信通过实战锻炼，你会快速掌握这些通用原则和最佳实践，形成自己的知识库。在遇见类似的场景时，可以灵活调取。

扩展篇：了解工具、生态和未来，开拓你的技术视野

我们学习 Hooks，是为了更好地使用 React，因此，在最后一个模块，我们必须要把视角扩展至 React 的整个生态。

我会给你介绍 React 应用该如何打包，整个生态有哪些可用的工具和资源，该如何测试，以及未来会有哪些发展的趋势。有了完整体系的积累，你才能真正独当一面，完成整个应用的开发、测试、打包和部署。

最后，希望这门从函数组件角度出发的 React Hooks 课程，不仅能够帮助你全面掌握 React 的开发技能，同时也能够给你一个很好的进阶起点，让你在 React 的开发路程上更加高效地前进。

当然，如果你在学习过程中有任何问题或者心得，也欢迎在留言区和我交流互动，我们一定可以共同进步。

© 版权归极客邦科技所有，未经许可不得传播售卖。页面已增加防盗追踪，如有侵权极客邦将依法追究其法律责任。