区块链开发指南_区块链开发权威指南

翻译

cumi7754 于 2020-07-19 07:58:10 发布 1076 收藏 2
文章标签： 编程语言 区块链 比特币 人工智能 java
原文链接：https://www.freecodecamp.org/news/the-authoritative-guide-to-blockchain-development-855ab65b58bc/

区块链开发指南

by Haseeb Qureshi

由Haseeb Qureshi

区块链开发权威指南 (The authoritative guide to blockchain development)

Cryptocurrencies, ICOs, magic internet money — it's all so damn exciting, and you, the eager developer, want to get in on the madness. Where do you start?

加密货币，ICO，神奇的互联网货币-真是太令人兴奋了，而您，热情的开发人员，也想抓狂。 从哪里开始？

I'm glad you're excited about this space. I am too. But you'll probably find it's unclear where to begin. Blockchain is moving at breakneck speed, but there's no clear onramp to learning this stuff.

我很高兴您对这个空间感到兴奋。 我也是。 但是您可能会发现不清楚从哪里开始。 区块链正以惊人的速度发展，但是学习这些东西并没有明显的障碍。

Since I left Airbnb to work full-time on blockchain, many people have reached out to me asking how to get into the blockchain space full-time. Consider this my authoritative (and inevitably incomplete) guide on how to get into blockchain engineering.

自从我离开Airbnb从事区块链全职工作以来，许多人联系我，询问如何全职进入区块链领域。 考虑一下这本关于如何进入区块链工程的权威(不可避免地是不完整的)指南。

This guide will proceed in ten parts:

本指南将分十部分进行：

Why should you learn blockchain development?

为什么要学习区块链开发？

Prerequisites

先决条件

The theoretical foundations of Bitcoin

比特币的理论基础

Building a blockchain yourself

自己建立区块链

Ethereum and smart contract programming

以太坊和智能合约编程

Smart contract security

智能合约安全

Taking off the training wheels

卸下训练轮

Building your own projects

建立自己的项目

Navigating the blockchain community

导航区块链社区

Getting a job

得到一份工作

为什么要学习区块链开发？ (Why should you learn blockchain development?)

Before I answer that question, let me first note: blockchain is a massively overvalued space right now. These prices are unsustainable, and a crash is definitely coming. This has all happened before, and will probably happen again. But if you work long-term in this space, you'll learn to shrug off prices. In the words of Emin Gun Sirer — prices are the least interesting part of cryptocurrencies. These are massively important technologies, and they are going to irrevocably change the world.

在回答这个问题之前，我首先要指出：区块链现在是一个被严重高估的空间。 这些价格是不可持续的，并且肯定会崩溃。 这一切以前都曾经发生过，并且可能会再次发生。 但是，如果您在这个领域长期工作，您将学会摆脱价格。 用Emin Gun Sirer的话来说-价格是加密货币中最不有趣的部分。 这些是非常重要的技术，它们将不可改变地改变世界。

If you're unsure, I can't tell you whether or not you should jump in. But I can tell you five reasons that convinced me to take the leap:

如果您不确定，我不会告诉您是否应该加入。但是我可以告诉您五个说服我迈出这一步的理由：

It's still early.

还早。

Bitcoin was invented 10 years ago, but the rate of innovation has only reached a fever pitch in the last couple of years, especially with the launch of Ethereum in 2015. Most of the new companies and ideas in this space have been built on top of Ethereum, which is still very immature.

比特币是10年前发明的，但是在过去的几年中，创新的速度才达到顶峰，尤其是在2015年推出以太坊时。这一领域的大多数新公司和创意都建立在以太坊，还很不成熟。

Even if you start now, you can realistically become a world-class expert within a few years. Most people just haven't been doing this that long, and it won't be that hard to catch up. Starting now would be analogous to deep learning experts who began studying the topic in the late 2000s.

即使您现在就开始，您也可以在几年内成为世界一流的专家。 大多数人只是没有花那么长时间，而且赶上它并不难。 从现在开始类似于在2000年代末开始研究该主题的深度学习专家。

2. This space doesn't have a strong talent funnel yet.

2. 这个空间还没有强大的人才渠道。

Most of the best and brightest students at universities are focusing on machine learning, web programming, or game development. Sure, blockchains are getting more sexy in the public discourse, but they're still a weird and subversive topic on which to stake your career.

大学中大多数最优秀和最聪明的学生都将注意力放在机器学习，Web编程或游戏开发上。 当然，区块链在公众讨论中正变得越来越性感，但是它们仍然是一个奇怪而具有颠覆性的话题，可以与您的职业相关。

Early on, blockchain was exclusively the realm of cypherpunks, paranoids, and weirdos. That's only recently begun to change. Just by being a curious and open-minded developer, you'll bring a lot of value to the space.

早期，区块链仅是密码，偏执狂和怪人的领域。 这只是最近才开始改变。 仅仅通过成为一个好奇和豁达的开发人员，您将为这个空间带来很多价值。

3. Much of the innovation is happening outside of academia.

3.许多创新发生在学术界之外。

Satoshi Nakamoto was not an academic as far as we know. There's no university or institution that offers a coherent blockchain concentration yet. Most of the innovation here has been led by aficionados, entrepreneurs, and independent researchers. Almost everything you need to know is in white papers, blog posts, public Slack channels, and open-source software. All it takes is rolling up your sleeves and jumping into the fray.

据我们所知，中本聪不是一位学者。 尚无大学或机构提供一致的区块链集中度。 这里的大多数创新都是由爱好者，企业家和独立研究人员领导的。 您需要了解的几乎所有内容都在白皮书，博客文章，公共Slack频道和开源软件中。 它所要做的就是卷起袖子，跳入战斗。

4. The demand for talent far, far exceeds supply.

4.对人才的需求远远超过了供给。

There just aren't enough developers in this space, and they can't get trained fast enough. Everyone is competing to hire blockchain talent, and projects are feeling the talent crunch. Many of the best companies can't pay their people enough to stay because they have too many opportunities. If you get some skills under your belt, it'll be easy to land a job.

在这个领域中，没有足够的开发人员，而且他们的培训速度不够快。 每个人都在竞争聘用区块链人才，项目感到人才紧缩。 许多最好的公司不能给他们的员工足够的薪水，因为他们拥有太多的机会。 如果您掌握了一些技能，就很容易找到工作。

5. Cryptocurrencies are just really damn cool.

5.加密货币真是太酷了。

Where else can you build sci-fi stuff like cryptographically secured, decentralized money? It's the wild west right now—and this brings good and bad. The space could use more transparency, and regulation will eventually come. But without a doubt, cryptocurrencies are one of the most innovative areas you can be working in right now.

您还可以在其他地方建立科幻小说之类的东西，例如加密安全的分散资金？ 现在是狂野的西部-带来好与坏。 这个空间可能会使用更多的透明度，并且最终将出现管制。 但毫无疑问，加密货币是您现在可以从事的最具创新性的领域之一。

Naval Ravikant said in a recent interview: the key to success is to give society things that it wants, but doesn't know how to get on its own. You can't go to school for such things; if you could, the world would already have a steady supply of it.

海军部长拉维坎特(Naval Ravikant) 在最近的一次采访中说：成功的关键是为社会提供它想要的东西，但不知道如何独立发展。 你不能因为这些事情而去学校。 如果可以的话，世界将已经有稳定的供应。

So build something no one else knows how to build. Right now, blockchains are brand new and there's so much left to figure out. If you succeed in building the future of decentralized technology, the world will reward you handsomely.

因此，构建任何其他人都不知道如何构建的东西。 现在，区块链是全新的，还有很多需要解决。 如果您成功构建了去中心化技术的未来，那么世界将为您带来丰厚的回报。

So say you want to throw in your hat. What do you need to know before you get into the ring?

所以说你想戴上帽子。 在进入电话圈之前，您需要了解什么？

先决条件 (Prerequisites)

I'd recommend strengthening up your understanding of fundamentals before you dive further. Blockchains are built atop decades of research in computer science, cryptography, and economics. Satoshi Nakamoto was a renegade, but he also knew well the history that preceded him. In order to understand why blockchains work, you need to understand their building blocks — what came before blockchains, and why those things didn't work.

我建议您在进一步潜水之前加深对基本原理的理解。 区块链是建立在数十年的计算机科学，密码学和经济学研究之上的。 中本聪是个叛徒，但他也很了解他之前的历史。 为了了解区块链为什么起作用，您需要了解它们的构件-区块链之前发生的事情以及这些事情为什么不起作用。

Here are some good prerequisites to be familiar with, in order of importance.

按照重要性的顺序，这里是一些熟悉的先决条件。

Note, these links are just a starting point, you'll probably want to dive deeper for many of these topics.

请注意，这些链接只是一个起点，您可能需要更深入地研究这些主题。

计算机科学 (Computer science)

数据结构 (Data structures)

You'll want to be familiar with the characteristics and complexity guarantees of the major data structures: linked lists, binary search trees, hash maps, and graphs (specifically, directed acyclic graphs which feature prominently in blockchains). It helps to have built them from scratch to better understand how they work and their properties.

您需要熟悉主要数据结构的特性和复杂性保证： 链表， 二进制搜索树， 哈希图和图 (特别是有向无环图，在区块链中占主导地位)。 从头开始构建它们有助于更好地了解它们的工作方式和属性。

密码学 (Cryptography)

Cryptography is the namesake and bedrock of cryptocurrencies. All cryptocurrencies use public/private key cryptography as the basis for identity and authentication. I'd recommend studying RSA (it's easy to learn, and doesn't require a very strong math background), then look at ECDSA. Elliptic curve cryptography requires significantly more abstract math — it's not important to understand all the details, but know that this is the cryptography that's used in most cryptocurrencies, including Bitcoin.

密码学是加密货币的代名词和基石。 所有加密货币都使用公共/专用密钥加密作为身份和身份验证的基础。 我建议您学习RSA (这很容易学习，并且不需要很强的数学背景)，然后再看ECDSA 。 椭圆曲线密码学需要更多的抽象数学知识-了解所有细节并不重要，但要知道这是大多数加密货币(包括比特币)中使用的密码学。

The other important cryptographic primitive is the cryptographic hash function. These can be used to construct commitment schemes, and are the building block for Merkle trees. Merkle trees enable Merkle proofs, one of the key optimizations that blockchains use for scalability.

另一个重要的密码原语是密码哈希函数 。 这些可以用来构建承诺方案 ，并且是Merkle树的构建块。 Merkle树启用Merkle证明 ， Merkle证明是区块链用于可伸缩性的关键优化之一。

分布式系统 (Distributed systems)

There are a few good textbooks on distributed systems, but it's a sprawling and difficult area of study. Distributed systems are absolutely essential to reasoning about blockchains, so you must build a foundation here before tackling blockchain programming.

关于分布式系统，有一些不错的教科书 ，但这是一个庞大且困难的学习领域。 分布式系统对于推理区块链绝对必不可少，因此在进行区块链编程之前，您必须在此处建立基础。

Once you're no longer living on a single machine, you have to start reasoning about consistency and consensus. You'll want to know the difference between linearizable and eventual consistency models. You'll also want to learn the guarantees of fault-tolerant consensus algorithms, such as Paxos and RAFT. Know the difficulties of reasoning about time in a distributed system. Appreciate the tradeoffs between safety and liveness.

一旦您不再生活在一台机器上，就必须开始进行有关一致性和共识性的推理。 您将要知道线性化模型和最终一致性模型之间的区别。 您还将需要学习诸如Paxos和RAFT之类的容错共识算法的保证。 了解分布式系统中时间推理的困难 。 理解安全性和活力之间的权衡。

With that background, you'll be able to understand the difficulties around Byzantine fault-tolerant consensus, the primary security requirement of public blockchains. You'll want to learn about PBFT, one of the first scalable algorithms to deliver Byzantine fault-tolerant consensus. PBFT is the basis for many non-proof-of-work blockchain consensus algorithms. Once again, you don't need to understand the details of how and why PBFT is correct, but get the general idea and its security guarantees.

在这样的背景下，您将能够理解围绕拜占庭容错共识 (公共区块链的主要安全要求)所遇到的困难。 您将要了解PBFT ，它是首批提供拜占庭容错共识的可扩展算法之一。 PBFT是许多非工作量证明区块链共识算法的基础。 再一次，您不需要了解PBFT如何正确以及为什么正确的细节，而是获得了总体思路及其安全性保证。

It's also very useful to understand the traditional methods of distributing databases (at its core, blockchains are databases after all). Learn about sharding (such as via consistent hashing), leader-follower replication, and quorum-based commits. Look into distributed hash tables (DHTs), such as Chord or Kademlia.

了解分布数据库的传统方法也非常有用(从根本上讲，区块链毕竟是数据库)。 了解分片 (例如，通过一致性哈希)， 领导者跟随者复制和基于仲裁的提交 。 查看分布式哈希表(DHT) ，例如Chord或Kademlia 。

联网 (Networking)

The decentralization of blockchains derives in large part from their peer-to-peer network topology. As such, blockchains are direct descendants of the past P2P networks.

区块链的去中心化很大程度上源于其对等网络拓扑。 因此，区块链是过去P2P网络的直接后代。

To understand the blockchain communication model, you need to understand the basics of computer networking: this means understanding TCP vs UDP, the packet model, what IP packets look like, and roughly how Internet routing works.

要了解区块链通信模型，您需要了解计算机网络的基础知识：这意味着了解TCP与UDP， 数据包模型，IP数据包的外观以及Internet路由的大致工作方式。

Public blockchains tend to spread messages via gossip protocols using flooding. It's instructive to learn the history of P2P network design, from Napster to Gnutella, BitTorrent and Tor. Blockchains have their own place, but they draw upon the lessons of these networks and how they were designed.

公共区块链倾向于使用洪泛通过八卦协议传播消息。 了解从Napster到Gnutella， BitTorrent和Tor的P2P网络设计的历史是很有启发性的。 区块链有其自己的位置，但它们借鉴了这些网络的经验教训以及它们的设计方式。

经济学 (Economics)

Cryptocurrencies are inherently multidisciplinary — this is part of what makes them so fascinating and radical. Besides computer science, cryptography, and networking, they are also deeply interwoven with economics. Cryptocurrencies can derive many security properties through their economic structures, which is often termed *cryptoeconomics*. As such, economics is essential to understanding cryptocurrencies.

加密货币本质上是多学科的，这是使其变得如此迷人和激进的原因之一。 除了计算机科学，密码学和网络，它们还与经济学紧密地交织在一起。 加密货币可以通过其经济结构获得许多安全属性，通常被称为*加密经济学*。 因此，经济学对于理解加密货币至关重要。

博弈论 (Game theory)

The most important branch of economics that plays into cryptocurrencies is game theory, the study of payoffs and incentives among multiple agents. You don't need to go *extremely* deep here, but you do need to understand the basic tools of game theoretic analysis and how you can use them to analyze incentives in one-shot and iterated games.

参与加密货币的经济学最重要的分支是博弈论，即对多主体之间收益和激励的研究。 你不需要去*极深*这里，但你需要了解博弈论分析的基本工具，以及如何使用它们来分析一次性奖励和迭代游戏。

Two key concepts in your repertoire should be Nash equilibria and Schelling points, as they feature prominently in cryptoeconomic analysis.

清单中的两个关键概念应该是Nash均衡和Schelling点，因为它们在加密经济分析中非常重要。

宏观经济学 (Macroeconomics)

Cryptocurrencies are not just protocols, they are also forms of money. As such, they respond to the laws of macroeconomics (if they can be called laws). Cryptocurrencies are subject to different monetary policies, and respond predictably to inflation and deflation. You should understand these processes and the effects they have on spending, saving, etc.

加密货币不仅是协议，而且还是货币形式。 因此，它们响应宏观经济学的法则(如果可以将其称为法则)。 加密货币受制于不同的货币政策，并且可以预料地应对通货膨胀和通货紧缩。 您应该了解这些过程及其对支出，储蓄等的影响。

Another valuable economic concept is the velocity of money, especially as it corresponds to valuing a currency.

另一个有价值的经济概念是货币的流动性，尤其是它对应于对货币进行估值时。

微观经济学 (Microeconomics)

Cryptocurrencies are also deeply interwoven with markets, which requires an understanding of microeconomics. You'll need a strong intuition for supply and demand curves. You should be able to reason about competition and opportunity costs (they'll apply frequently to cryptocurrency mining). For many coin distributions and cryptoeconomic systems, auction theory features prominently.

加密货币也与市场紧密地交织在一起，这需要对微观经济学的理解。 您需要对供需曲线有很强的直觉。 您应该能够推断出竞争和机会成本 (它们会经常应用于加密货币挖矿)。 对于许多硬币发行和加密经济系统， 拍卖理论具有突出的特征。

I expect you'll be familiar with some of these topics already. If you are, feel free to skim or skip over them entirely.

我希望您已经熟悉其中一些主题。 如果您愿意，可以随意浏览或完全跳过它们。

Okay, by now you've gone through and shored up your fundamentals (or maybe you skipped a bunch, who's counting?), so now that you've got your theory in check, let's get started on blockchain development.

好的，到目前为止，您已经遍历并巩固了基础知识(或者您跳过了一堆，谁在指望？)，所以现在您已经掌握了理论，让我们开始进行区块链开发。

比特币的理论基础 (The Theoretical Foundations of Bitcoin)

In October of 2008, Satoshi Nakamoto published a white paper in which he described a protocol for a decentralized digital currency. He called this protocol Bitcoin.

2008年10月，中本聪(Satoshi Nakamoto)发表了一份白皮书，其中他描述了一种去中心化数字货币的协议。 他称该协议为比特币。

Before you can understand the big ideas behind blockchains, you have to start with Bitcoin and grasp Satoshi's original insight.

在您了解区块链背后的大创意之前，您必须从比特币开始，并掌握Satoshi的原始见解。

First, I recommend building your intuitions about proof-of-work and the fork choice rule (also known as Nakamoto consensus). Start here:

首先，我建议您建立关于工作量证明和分叉选择规则(也称为Nakamoto共识)的直觉。 从这里开始：

I recommend watching more than one video explanation to get the idea seared into your head:

我建议观看不止一个视频说明，以使您的想法浮出水面：

Great. Now that you've built up your intuition, this article will provide a deeper end-to-end exposition of the critical components of how Bitcoin works.

大。 现在您已经建立了自己的直觉， 本文将提供有关比特币工作原理的重要组成部分的更深入的端到端说明。

自己建立区块链 (Building a blockchain yourself)

Now that you have the high-level intuition, it's time to build your own proof-of-work based blockchain. Don't worry, it's easier than it sounds. Here are some good resources.

现在，您已经有了高级的直觉，现在该构建自己的基于工作量证明的区块链了。 别担心，这比听起来容易。 这里有一些很好的资源。

First, I have a video lecture where I walk through exactly how to do this in Ruby (I recommend watching even if you're not a Ruby programmer):

首先，我有一个视频讲座，其中我将详细介绍如何在Ruby中执行此操作(即使您不是Ruby程序员，我也建议您观看)：

Source and slides here.

来源和幻灯片在这里。

There are also other blockchain implementations you can find, written in various programming languages. Go on and build your own, and satisfy yourself that it's mostly functional.

您还可以找到其他用各种编程语言编写的区块链实现 。 继续构建您自己的，并让您满意，它基本上可以正常工作。

Once you've made it this far, you should have a good grasp of how to implement a simple payments application atop a blockchain (i.e., Bitcoin). You should also by now have enough background that you should be able to read and understand the original Bitcoin whitepaper.

一旦到此为止，您应该对如何在区块链(即比特币)上实现简单的支付应用程序有一个很好的了解。 现在，您还应该具有足够的背景知识，以便能够阅读和理解原始的比特币白皮书 。

To understand the economics and mechanics of Bitcoin mining, I recommend watching the lecture on Bitcoin mining in the Bitcoin and Cryptocurrencies Princeton course.

为了了解比特币挖矿的经济学和原理，我建议在"比特币和加密货币普林斯顿"课程中观看有关比特币挖矿的讲座 。

If you've gotten this far, you should understand Bitcoin well enough to walk through a Bitcoin block header and understand what each of its components mean. You should also be able to play around with a Bitcoin block explorer and navigate raw Bitcoin transactions.

如果到此为止，您应该对比特币有足够的了解，以遍历比特币区块头并了解其每个组成部分的含义。 您还应该能够与比特币区块浏览器一起玩，并浏览原始比特币交易。

Now is a good time to study up on the history of Bitcoin and cryptocurrencies. The below video, offered by a UC Berkeley Decal, gives a good overview.

现在是学习比特币和加密货币历史的好时机。 下面的视频由UC Berkeley Decal提供，提供了很好的概述。

Some more extra credit resources:

一些额外的信贷资源：

> Academic precursors to Bitcoin
>
> 比特币的学术先驱
>
> Mechanics of Bitcoin: UTXOs and Bitcoin script (Bitcoin script is not super important, just know roughly what it can do)
>
> 比特币的机制： UTXO和比特币脚本 (比特币脚本不是非常重要，只是大致了解它可以做什么)
>
> Short guide to Bitcoin forks
>
> 比特币叉子的简短指南

Soft forks and miner signaling

软叉和矿工信号

Double spends, 51% attacks, and selfish mining

双花，51％攻击和自私的挖矿

Replay attacks

重播攻击

Bitcoin scalability problems, which is the source of most of the contentiousness in the Bitcoin ecosystem. You should have an idea of why Bitcoin folks argue so much about the block size.

比特币可扩展性问题，这是比特币生态系统中大多数争议的根源。 您应该对比特币人士为何对区块大小这么多争论有个想法。

Segregated witness, a.k.a. SegWit, not essential but it comes up a lot.

孤立的见证人，又名SegWit，不是必需的，但出现了很多。

Lightning Network, one of the more important scaling solutions for Bitcoin, also generalizes to other blockchains

闪电网络 (Lightning Network)是比特币更重要的扩展解决方案之一，也可以推广到其他区块链

Bitcoin full nodes, Bitcoin fee statistics, charts, charts and more charts

比特币完整节点， 比特币费用统计， 图表， 图表和更多图表

Bitcoin energy consumption index (at the time of publication, Bitcoin mining consumes as much energy as all of Peru)

比特币能源消耗指数 (在发布之时，比特币采矿所消耗的能源与秘鲁全境一样多)

Insightful essay by Gwern on the scrappy inelegance of Bitcoin

Gwern的有见地的文章关于比特币的草率的优雅

Jameson Lopp has a wealth of other resources on Bitcoin if you want to go deeper down the rabbit hole.

如果您想深入研究兔子漏洞，Jamesson Lopp拥有大量的比特币其他资源 。

以太坊和智能合约编程 (Ethereum and smart contract programming)

Now that you've built a blockchain and understand the dynamics of Bitcoin, it's time to delve into Ethereum.

现在您已经建立了区块链并了解了比特币的动态，是时候深入研究以太坊了。

You understand how blockchains and proof-of-work can achieve distributed, Byzantine fault-tolerant consensus within a peer-to-peer network. But a payments network is just one application you can run atop such a blockchain. In 2013, Vitalik Buterin, the creator of Ethereum asked: what if you used a blockchain to implement a decentralized computer?

您了解了区块链和工作量证明如何在对等网络中实现分布式拜占庭容错共识。 但是支付网络只是您可以在这样的区块链之上运行的一个应用程序。 2013年，以太坊的创建者Vitalik Buterin问：如果您使用区块链来实现去中心化计算机该怎么办？

In Ethereum, you pay miners to execute your programs on this distributed virtual machine. This means you can perform arbitrary computations, using a Turing-complete programming language (unlike Bitcoin script). Obviously that includes payments-related applications, so Ethereum enables a superset of Bitcoin's functionality and has birthed a renaissance of innovation.

在以太坊中，您需要向矿工付费以在此分布式虚拟机上执行程序。 这意味着您可以使用图灵完备的编程语言(不同于比特币脚本)执行任意计算。 显然，其中包括与支付相关的应用程序，因此以太坊启用了比特币功能的超集，并催生了创新复兴。

This brings us to smart contracts — the name for programs that run on such a virtual machine. A smart contract can interact directly with the blockchain's cryptocurrency in accordance with the execution of a program. In other words, you can create financial contracts that automatically enforce themselves. It's a wild idea, and all sorts of sci-fi futuristic stuff you can do once you embrace this programming model.

这将我们带入智能合约-在这样的虚拟机上运行的程序的名称。 智能合约可以根据程序的执行直接与区块链的加密货币进行交互。 换句话说，您可以创建自动执行的金融合同。 这是一个疯狂的想法，一旦您采用了这种编程模型，便可以进行各种科幻的未来派工作。

Ethereum has enabled the wave of ICOs and developers building atop the blockchain. It is the second largest cryptocurrency behind Bitcoin, it has more than 10x the developers of the next most popular platform, it has the strongest developer team, the most mature tooling, and the majority of ICOs and projects atop it. It also has the most industry support, which goes a long way. In all likelihood, if you're doing blockchain development, you'll be writing code for Ethereum smart contracts. (Even if you're not, it's essential to understanding what's going on in this space.)

以太坊推动了ICO和开发人员在区块链之上构建的浪潮。 它是仅次于比特币的第二大加密货币，它的开发者人数是第二受欢迎的平台的开发者的十倍以上，拥有最强大的开发人员团队，最成熟的工具以及其上的大多数ICO和项目。 它还拥有最多的行业支持，这将有很长的路要走。 很有可能，如果您正在进行区块链开发，那么您将为以太坊智能合约编写代码。 (即使您不是，也必须了解该空间中发生的事情。)

First, a more detailed high-level explanation of Ethereum:

首先，对以太坊进行更详细的高级解释：

The ideas behind Ethereum have also spawned a wave of innovation in cryptoeconomics. You should dip your toes into the ideas around DAOs, and all of the sci-fi fever dreams that they hint at.

以太坊背后的思想也催生了密码经济学的创新浪潮 。 您应该全神贯注于DAO的想法，以及它们暗示的所有科幻发烧梦想。

Okay, that's enough fantasy, let's dig into the tech.

好吧，那就足够了，让我们深入研究技术。

Here's a good overview of the Ethereum yellow paper and its internals, by Preethi Kasireddy. Ethereum uses an account model rather than Bitcoin's UTXO model — you'll soon see why this makes it easier to write smart contracts.

这是Preethi Kasireddy 对以太坊黄纸及其内部的很好概述 。 以太坊使用账户模型而不是比特币的UTXO模型-您将很快明白为什么这使得编写智能合约变得更加容易。

As with any technology, the best way to get acquainted with Ethereum is by building a few small projects.

与任何技术一样，了解以太坊的最佳方法是建立一些小项目。

The dominant programming language for Ethereum is Solidity, which is a statically typed JavaScript-esque language. It's a language with a lot of warts, and many questionable design choices. More robust languages like Viper may replace it once they're production-ready, but for now Solidity is the lingua franca of smart contract programming. It's basically Ethereum's JavaScript, so you'll need to learn it (and its pitfalls).

以太坊的主要编程语言是Solidity，它是一种静态类型JavaScript类语言。 这是一种有很多疣的语言，并且有很多可疑的设计选择。 Viper等更健壮的语言一旦投入生产，就可以取代它，但就目前而言，Solidity是智能合约编程的通用语言。 它基本上是以太坊JavaScript，因此您需要学习它(及其陷阱)。

To get your first exposure to Solidity development, I recommend working through all of the CryptoZombies tutorial. It's a delightful and high-quality Codecademy-esque tutorial that will teach you the basics of Solidity programming.

为了使您第一次接触Solidity开发，建议您阅读所有CryptoZombies教程 。 这是一个令人愉悦的高质量Codecademy风格的教程，它将教您Solidity编程的基础知识。

Now that you've whetted your appetite, it's time to develop on your own.

现在，您已经激发了胃口，现在该是您自己发展的时候了。

The "hello world" of Ethereum is building an ERC-20 compliant token. I recommend this guide as a first tutorial to walk you through the process.

以太坊的" hello world"正在构建符合ERC-20的令牌 。 我建议将此指南作为第一个教程，以逐步指导您完成该过程。

Remix is an in-browser Solidity editor and compiler — it's basically the training wheels of Ethereum development, so I recommend working through the rest of your practice in Remix. But it's also worth setting up a local blockchain and getting a sense of the Ethereum tooling. This tutorial does a good job of walking you through an end-to-end blockchain stack and explaining the pieces as they go along.

Remix是一个浏览器内部的Solidity编辑器和编译器-基本上，它是Ethereum开发的训练轮，所以我建议您在Remix中完成其余的练习。 但是，建立本地区块链并了解以太坊工具也是值得的。 本教程很好地引导您完成了端到端的区块链堆栈，并解释了各个过程。

Next I'd recommend building a voting system. I'd call this the Todo App of Ethereum. Karl Floersch has a great tutorial where he walks through how to build a secure commit-reveal voting system.

接下来，我建议构建一个投票系统。 我将其称为以太坊的Todo App。 Karl Floersch有一个很棒的教程 ，其中他逐步介绍了如何构建安全的提交-公开投票系统。

Great, now for your mid-term exam: build a secure coin toss game, where two players can securely bet on the coin flip. No tutorial this time, do it on your own. Think about possible attacks — how can the players cheat? Can you ensure that they play honestly? Here are some hints.

太好了，现在适合您的期中考试：构建一个安全的掷硬币游戏，两个玩家可以安全地在掷硬币游戏上下注。 这次没有教程，请自行完成。 考虑一下可能的攻击-玩家如何作弊？ 您能确保他们诚实地比赛吗？ 这里有一些提示 。

智能合约安全 (Smart contract security)

Security is absolutely essential to blockchain development. Smart contracts have been plagued by disastrous hacks, including the DAO hack, the Parity Wallet hack, and the affectionately named Parity Wallet hack 2 (now with its own T-shirt). You absolutely must read analyses of all three of these hacks if you're going to be writing production smart contracts.

安全性对于区块链开发绝对至关重要。 智能合约受到灾难性黑客的困扰，包括DAO黑客， Parity Wallet hack和亲切地命名为Parity Wallet hack 2 (现在带有自己的T恤)。 如果要编写生产智能合约，则绝对必须阅读对所有这三种黑客的分析。

The truth is, **smart contracts are extremely hard to get right**. Though the programming toolchain will improve to make these exact attacks harder, they were ultimately all due to programmer error. There are also many subtler bugs that arise from smart contract programming, such as in frontrunning or secure generation of randomness.

事实是， **智能合约很难正确地制定** 。 尽管编程工具链将有所改进，以使这些确切的攻击更加困难，但最终归因于程序员的错误。 智能合约编程还会产生许多微妙的错误，例如在前端运行或安全地生成随机性 。

As a smart contract developer, you must treat security as paramount. There's no "move fast and break things" in smart contract programming. That means any code that handles significant flows of money should be run through static analyzers like Oyente or Securify, tested thoroughly, and then audited by an experienced smart contract auditor. You should also try to rely on pre-audited components, such as OpenZeppelin's open source contracts.

作为智能合约开发人员，您必须将安全性视为重中之重。 智能合约编程中没有"快速行动并打破常规"。 这意味着任何代码来处理的钱显著流应该通过像静态分析仪运行Oyente或Securify ，彻底的测试，然后由经验丰富的智能合约核数师审计。 您还应该尝试依赖预先审核的组件，例如OpenZeppelin的开源合同 。

To strengthen your security chops, I recommend working through The Ethernaut by OpenZeppelin, a game where you find and attack vulnerabilities in smart contracts. Many of them have you replicate real attacks against smart contracts that have occurred in the wild.

为了加强您的安全性，我建议您通过OpenZeppelin开发的The Ethernaut，该游戏可在其中找到并攻击智能合约中的漏洞。 他们中的许多人都可以复制针对野外发生的智能合约的真实攻击。

Phil Daian also has an excellent set of smart contract hacking challenges called Hack This Contract.

菲尔·戴安(Phil Daian)还拥有一系列出色的智能合约黑客攻击难题，称为" 破解此合约" 。

Once you make it past that, I strongly recommend reading the entirety of Smart Contract Best Practices, compiled by ConsenSys. Expect to revisit this document many times over in your smart contract programming career. The bibliography is also worth exploring for further reading by security experts.

一旦您做到这一点，我强烈建议您阅读由ConsenSys编制的《 智能合约最佳实践》的全部内容。 期望在您的智能合约编程生涯中多次访问此文档。 参考书目也值得探索，以供安全专家进一步阅读。

卸下训练轮 (Taking off the training wheels)

If you've made it this far, you should now be ready to move past Remix and start using a serious Solidity development stack.

如果到此为止，您现在应该准备通过Remix，开始使用认真的Solidity开发堆栈。

Most developers recommend VSCode or Atom for your text editor, since they have decent Solidity plugins. For interacting with a local blockchain, you'll want to use Ganache (formerly TestRPC), and you'll want to use the Truffle framework for your (JS-based) tests and configuring your build pipeline.

大多数开发人员都建议将VSCode或Atom用于文本编辑器，因为它们具有不错的Solidity插件。 为了与本地区块链进行交互，您将要使用Ganache (以前称为TestRPC)，并且要使用Truffle框架进行基于JS的测试并配置构建管道。

Now is a good time to look into IPFS, which you can use as a fully decentralized filestore at much cheaper cost than the Ethereum blockchain. Here's a brief explainer by the creator, Juan Benet:

现在是研究IPFS的好时机，您可以将其用作完全去中心化的文件存储，而成本比以太坊区块链便宜得多。 这是创作者Juan Benet的简短解释：

For interacting with Ethereum and IPFS full nodes, Infura is what most devs recommend. Etherscan and ETH Gas Station provide useful real-time stats on the Ethereum network.

为了与以太坊和IPFS完整节点进行交互，大多数开发人员建议使用Infura 。 Etherscan和ETH加油站在以太坊网络上提供有用的实时统计信息。

Once you have your full Web3 stack set up, try deploying an end-to-end Dapp (decentralized application). This tutorial provides a nice full-stack overview using Node and Postgres for the backend, and this tutorial will show you how to create a fully decentralized application, using IPFS as your persistence layer.

设置完完整的Web3堆栈后，请尝试部署端到端Dapp(去中心化应用程序)。 本教程提供了一个很好的全栈概述，其中使用Node和Postgres作为后端，并且本教程将向您展示如何使用IPFS作为持久层来创建完全去中心化的应用程序。

建立自己的项目 (Building your own projects)

You should now be comfortable with most of the tech — what's left is to start building stuff and going deeper into the blockchain community.

您现在应该对大多数技术都感到满意-剩下的就是开始构建东西并更深入地进入区块链社区。

First, start building your own projects. If there's some great idea that you're excited about, go build it, and convince others to hack on it with you! If you don't have an idea yet or aren't comfortable getting your hands dirty, there are many high-quality open source projects that welcome contributions. OpenZeppelin might be a good place to start for smart contracts.

首先，开始构建自己的项目。 如果您对某个伟大的想法感到兴奋，那就去构建它，并说服其他人与您一起破解它！ 如果您还没有一个主意或者不满意，那么有很多高质量的开源项目都欢迎您的参与。 OpenZeppelin可能是开始智能合约的好地方。

Better yet, I'd recommend starting by finding an actively developed project that you're a fan of. Get on their Slack or Rocketchat — the devs are usually readily accessible. Tell them you'd like to contribute and ask for some small tasks (or find unresolved issues on their Github).

更好的是，我建议从找到一个您喜欢的积极开发的项目开始。 使用Slack或Rocketchat-开发人员通常很容易获得。 告诉他们您想贡献并要求一些小任务(或在他们的Github上找到未解决的问题)。

Note that while I've been focusing on protocols and smart contract development, blockchain companies need web developers to build their core functionality. These roles will often require interacting with blockchains, so it's essential to have a good mental model of how blockchains work — but for many engineers at blockchain startups, most of your work will be in building a Python webserver, or designing a React frontend, and interacting with the blockchain may be a small part of that job. You don't have to specialize in smart contract development — in reality, that's only one part of a working blockchain stack.

请注意，尽管我一直专注于协议和智能合约开发，但区块链公司需要Web开发人员来构建其核心功能。 这些角色通常需要与区块链进行交互，因此拥有一个关于区块链如何运作的良好思维模型至关重要–但是对于区块链初创公司的许多工程师来说，您的大部分工作将是构建Python网络服务器或设计React前端，与区块链进行交互可能只是该工作的一小部分。 您不必专门研究智能合约开发-实际上，这只是工作中的区块链堆栈的一部分。

Beyond open source contributions, there are also many blockchain hackathons constantly popping up. Most projects have a free public Slack you can join, and there's a very active Gitter channel for Ethereum itself where lots of devs hang out. As you go deeper into the space, you'll eventually find your peer group, whether it be in a Slack channel, Telegram group, or Gitter channel. Wherever it is, find your people and continue learning.

除了开源之外，还有许多区块链黑客马拉松不断涌现。 大多数项目都有一个免费的公共Slack，您可以加入，并且有一个非常活跃的以太坊本身的Gitter渠道，许多开发人员都在这里闲逛。 当您深入该空间时，最终将找到您的对等组，无论是在Slack通道，Telegram组还是Gitter通道中。 无论身在何处，找到您的员工并继续学习。

导航区块链社区 (Navigating the blockchain community)

The best way to really understand the blockchain world is to immerse yourself in it. Read and listen to the smartest people, especially stuff they've written in the past. This has always been my strategy when trying to learn a new domain, and it's paid dividends for me.

真正了解区块链世界的最好方法是将自己沉浸在其中。 阅读和聆听最聪明的人，尤其是他们过去写的东西。 在尝试学习新领域时，这一直是我的策略，这对我来说是有好处的。

There's lots of good blockchain content out there, but there's also a lot of crap. Here's the information diet I recommend.

有很多好的区块链内容，但是也有很多废话。 这是我推荐的信息饮食。

媒体 (Media)

The three fantastic podcasts I recommend are the Software Engineering Daily Blockchain interviews, which provide good technical intros to many topics and cryptocurrencies. From there I recommend Epicenter and Unchained — you'll want to go back and listen to many of the older episodes. Another interesting up-and-coming technical podcast is Conspiratus. I'd recommend subscribing to each of these.

我推荐的三个出色的播客是《 软件工程日报》区块链访谈，这些访谈为许多主题和加密货币提供了很好的技术介绍。 从那里，我推荐Epicenter和Unchained-您将要回去听许多旧的情节。 另一个有趣的新兴技术播客是Conspiratus 。 我建议您订阅每一个。

There are a few good Youtube channels (though there's tons of trash on Youtube). Subscribe to the Ethereum Foundation and watch Devcon3 presentations. Blockchain at Berkeley records many of their lectures, most of which are excellent technical overviews. Decypher Media also posts talks, whitepaper reviews, and tutorials. Jackson Palmer has engaging weekly overviews, these are on the less technical side but very evenly presented.

YouTube上有一些不错的频道(尽管YouTube上有很多垃圾)。 订阅以太坊基金会并观看Devcon3演示。 伯克利的区块链记录了他们的许多演讲，其中大多数都是出色的技术概述。 Decypher Media还发布演讲，白皮书评论和教程。 杰克逊·帕尔默(Jackson Palmer)每周进行一次引人入胜的概述，这些概述的技术性程度较低，但呈现得非常均匀。

在线阅读 (Online reading)

For realtime blockchain chatter, it lives mostly in two places: Reddit, and Twitter. For Reddit, most subreddits are very low quality and dominated by noise. r/Ethereum is consistently decent quality (and there are a few okay subreddits for specific cryptocurrencies). Most subreddits though are primarily dominated by speculators, and are not a good use of your attention. Stay away from Bitcoin-related subreddits. Bitcoin notoriously has one of the most toxic communities, and Reddit only magnifies that.

对于实时区块链聊天，它主要存在于两个地方：Reddit和Twitter。 对于Reddit而言，大多数子Reddit的质量都非常低，并且受到噪声的支配。 r /以太坊的质量始终如一(对于特定的加密货币，还有一些可以接受的分类)。 但是，大多数子市场主要由投机者主导，并且不能很好地利用您的注意力。 远离与比特币有关的次级债券。 众所周知，比特币是最有毒的社区之一，而Reddit仅将其放大。

Twitter is more of a mixed bag. For better or for worse, most blockchain people live on Twitter. Blockchain Twitter was somewhat of a mystery to me at first, but eventually I developed an informal ontology of Twitter blockchain people. From my experience, there are five types of blockchain personalities: the builders, the entrepreneurs, the journalists, the traders, and the “thought leaders.”

Twitter的情况更加复杂。 无论好坏，大多数区块链人都生活在Twitter上。 最初，区块链推特对我来说还是个谜，但最终我开发了推特区块链人的非正式本体。 根据我的经验，有五种类型的区块链个性：建设者，企业家，新闻记者，商人和“思想领袖”。

Avoid “thought leaders” like the plague. Entrepreneurs can be okay, though they mostly act as hype men or tweet about their own projects. Investors mostly tweet about prices and hype-y projects, so if that’s your thing, that’s your thing. Journalists tend to tweet about major news items of the day—I recommend staying away unless you need real-time analysis, which you probably don’t. If you’re an active trader it might be important, but if you’re trying to build on the blockchain, most real-time stuff is a distraction.

避免像瘟疫这样的“思想领袖”。 企业家可以没事，尽管他们大都充当炒作或在推销自己的项目。 投资者大多在推特上谈论价格和炒作项目，所以如果那是你的事，那是你的事。 记者倾向于发布当天的重大新闻-我建议您不要使用，除非您需要实时分析，而您可能不需要。 如果您是一位活跃的交易员，这可能很重要，但是如果您尝试建立在区块链上，那么大多数实时内容都会让人分心。

Pay the most attention to the builders. They’re the people who matter most right now, and who are pushing the technology forward.

最注意建筑商。 他们是目前最重要的人，他们正在推动技术向前发展。

A few representatives from each category (do a breadth-first search of who these people follow if you want to fill out your Twitter feed):

每个类别的一些代表(如果要填写您的Twitter feed，请广度优先搜索这些人的追踪对象)：

建筑商 (Builders)

Vitalik Buterin, Ethereum

以太坊(Vitalik Buterin)

Zooko Wilcox, ZCash

佐科· 威尔科克斯 (Zooko Wilcox)

Nick Szabo, inventor of smart contracts

智能合约的发明者Nick Szabo

Vlad Zamfir, Ethereum

以太坊(Vlad Zamfir)

Marco Santori, Cooley LLP

马可·圣托里 (Marco Santori)

Riccardo “fluffypony” Spagni, Monero

Riccardo“ fluffypony” Spagni，门罗

Matt Liston, Gnosis

马特·利斯顿

企业家 (Entrepreneurs)

Balaji Srinivasan, Earn.com

Balaji Srinivasan， Earn.com

Erik Voorhees, Shapeshift

埃里克·沃希尔斯 (Erik Voorhees)

投资人 (Investors)

Naval Ravikant, MetaStable

海军拉维坎特

Ari Paul, Blocktower Capital

Blocktower Capital的Ari Paul

Linda Xie, Scalar Capital

琳达谢，标量资本

Chris Burniske, Placeholder

占位符Chris Burniske

记者 (Journalists)

Tuur Demeester, Adamant Research

阿达曼特研究公司Tuur Demeester

Laura Shin, Forbes

福布斯》(Laura Shin)

(You should also follow me, though I definitely don’t belong on this list.)

(您也应该关注我，尽管我绝对不属于此列表。)

All that said, I recommend minimizing your exposure to Twitter and Reddit. If you’re not a journalist or a daytrader, chances are, you don’t need a firehose of realtime chatter. Important information will bubble up to you asynchronously. There are several good news digests that will summarize the most important news of the day/week that you can consume on your own time without being at the mercy of attention markets.

话虽如此，我建议您尽量减少对Twitter和Reddit的访问。 如果您不是新闻记者或日间交易员，那么您很有可能不需要实时聊天。 重要信息将异步地传给您。 有几个好消息摘要，将总结您可以在不依赖关注市场的情况下自行花费的一天/一周中最重要的新闻。

I recommend subscribing to Inside Bitcoin for daily digests of the most important crypto news pieces (it covers more than just Bitcoin). For token projects, Token Economy has excellent weekly writeups, and Week in Ethereum has good digests of developer-focused happenings in the Ethereum ecosystem.

我建议订阅Inside Bitcoin，以获取最重要的加密新闻的每日摘要(不仅涵盖比特币)。 对于代币项目，代币经济每周都有出色的记录，而以太坊周刊则很好地总结了以太坊生态系统中以开发人员为中心的事件。

Beyond this, you probably don't need to be monitoring for real-time news. Focus on building stuff and learning.

除此之外，您可能不需要监视实时新闻。 专注于构建东西和学习。

You'll want to follow the best blogs. Long-form content tends to be the best bang for the buck. I recommend following these:

您将要关注最好的博客。 长篇幅的内容往往是最好的选择。 我建议遵循以下这些：

- Vitalik Buterin for excellent blockchain and cryptoeconomic analysis (read all of his older blog posts as well, Vitalik is widely regarded as a once-in-a-generation thinker)
- Vitalik Buterin进行出色的区块链和加密经济分析(也请阅读他所有较早的博客文章，Vitalik被广泛认为是千载难逢的思想家)
- Hacking, Distributed for blockchain security analyses by Cornell researchers
- 黑客，由康奈尔大学研究人员分发用于区块链安全性分析
- Unenumerated, Nick Szabo's luminary blog with challenging and eclectic essays about the role of cryptocurrencies in society
- 尼克·扎博 (Nick Szabo)的名人博客未列举，关于加密货币在社会中的作用具有挑战性和折衷主义的文章
- Money Stuff, Matt Levine's Bloomberg syndication, with cutting and insightful analysis that touches on the intersection of markets, finance, and blockchain news
- Money Stuff，Matt Levine的Bloomberg联合组织，具有切入点和有见地的分析，涉及市场，金融和区块链新闻的交集
- Vlad Zamfir for tempered and cautious perspectives on the state and public blockchains
- Vlad Zamfir对州和公共区块链持谨慎态度
- Chris Burniske for a string of excellent blog posts on how to value crypto assets
- 克里斯·伯尼斯克 (Chris Burniske)撰写了一系列有关如何评估加密资产的优秀博客文章
- Jameson Lopp for his great technical posts from the perspective of a software engineer building for the blockchain ecosystem
- Jameson Lopp从为区块链生态系统构建软件工程师的角度发表了出色的技术文章
- Great Wall of Numbers by Tim Swanson, for his sober and unflinching deconstruction of blockchain mania, especially in the enterprise space
- 蒂姆·斯旺森(Tim Swanson)撰写的《万里长城》，他对区块链狂热的清醒和坚定的解构，尤其是在企业领域

(You should also read my blog, though again, I don't quite belong on this list.)

(您也应该阅读我的博客 ，尽管再次，我并不完全属于此列表。)

书籍和课程 (Books and courses)

If you want a more structured approach to learning this material, there are a few high-quality books and courses out there (and a lot of low-quality ones).

如果您想采用一种更结构化的方法来学习本文，那么这里有一些高质量的书籍和课程(还有很多低质量的书籍和课程)。

The best overall textbook for blockchains is Bitcoin and Cryptocurrency Technologies (which accompanies the Princeton Coursera course). The only other books I'd recommend in this space are Mastering Bitcoin by Andreas Antonopoulos, and his upcoming Mastering Ethereum, co-authored by Ethereum cofounder Gavin Wood (both published by O'Reilly). The one nontechnical book I'd recommend is Digital Gold by Nathaniel Popper. Pretty much everything else worth reading will be in blogs, not books — this space is moving so fast that the most important figures seldom have the time to write books, and books are often outdated by the time they're released.

最好的整体区块链教科书是比特币和加密货币技术 (普林斯顿Coursera课程随附)。 在这个领域，我唯一推荐的其他书籍是Andreas Antonopoulos的Mastering Bitcoin ，以及即将推出的Mastering Ethereum ，该书由以太坊联合创始人Gavin Wood合着(均由O'Reilly出版)。 我会推荐的一本非技术性书籍是Nathaniel Popper的Digital Gold 。 几乎所有其他值得阅读的内容都将出现在博客中，而不是书籍中-这种空间发展得如此之快，以至于最重要的人物很少有时间去写书，而书的发行时间往往已经过时了。

If you want a more structured approach to learning this material, there are a few high-quality courses out there (and a lot of low-quality ones). I've already linked to a couple lectures from the Princeton Coursera Course (the videos are on Youtube as well), and the UC Berkeley Decal. I've also heard good things about Consensys Academy for folks who want to get into smart contract development.

如果您想采用一种更有条理的方法来学习本文，那么这里有一些高质量的课程(还有很多低质量的课程)。 我已经链接到普林斯顿Coursera课程的一些讲座(视频也在Youtube上)和UC Berkeley Decal上 。 我还听说过Consensys学院的好事，适合想要进行智能合约开发的人们。

I'm also teaching a 4-week seminar on cryptocurrencies for software developers at the Bradfield School of Computer Science in SF. The course is in-person in SF only and seats are limited, since it's a small and in-depth seminar-style class. But if you're a software engineer in SF and want to learn more about the theory and practice behind cryptocurrencies, it might be a good fit for you.

我还在SF的Bradfield计算机科学学院为软件开发人员讲授为期4周的加密货币研讨会 。 由于这是一门小型且深入的研讨会式课程，因此该课程仅在SF中进行，并且席位有限。 但是，如果您是SF的软件工程师，并且想了解有关加密货币背后的理论和实践的更多信息，那么它可能非常适合您。

得到一份工作 (Getting a job)

As I said before, blockchain startups are hiring like crazy. If you've actually gotten this far and have done even half the things I suggested, you are probably already employable in this space. AngelList did a great writeup on how to get a job in the crypto space.

如我之前所说，区块链初创公司正在疯狂招募。 如果您实际上已经走了这么远，并且完成了我建议的一半工作，那么您可能已经在这个领域中受雇了。 AngelList在如何获得加密货币领域的工作方面写得很棒 。

There are several good aggregators for blockchain-related job postings:

有一些很好的聚合器可用于与区块链相关的职位发布：

- AngelList crypto startups

AngelList加密创业公司

BlockchainJobz

区块链乔布斯

Ethereum Jobs

以太坊工作

Be in Crypto

处于加密状态

Blockchain Job Board

区块链工作委员会

Crypto Jobs List

加密作业列表

Google jobs (blockchain search query)

Google职位(区块链搜索查询)

ConsenSys jobs (Ethereum venture studio with many projects under their umbrella)

ConsenSys职位 (以太坊风险工作室，旗下有许多项目)

Some particularly promising blockchain startups that I know are hiring devs:

我知道一些特别有前途的区块链初创公司正在招聘开发人员：

0x

0x

Dharma Labs

佛法实验室

Civic

思域

There are also a number of larger companies in the market for crypto devs:

市场上还有一些较大的加密开发人员公司：

Coinbase, the Google of crypto, is always hiring like crazy

加密货币的Google Coinbase一直疯狂地招聘

Stellar and Ripple if you want to work directly on more enterprise-friendly cryptocurrencies

恒星和纹波，如果你想直接工作在更多的企业友好cryptocurrencies

Square has integrated some blockchain, though not sure if they’re hiring externally

Square has integrated some blockchain, though not sure if they're hiring externally

IBM, Visa, or JP Morgan if you want to kick it old school

IBM , Visa , or JP Morgan if you want to kick it old school

(Note that this specific company list is super Bay Area-centric, because that's where I live, so your mileage may vary. The job aggregators are more global though.)

(Note that this specific company list is super Bay Area-centric, because that's where I live, so your mileage may vary. The job aggregators are more global though.)

But to my mind, the best way to get involved in a company is to find a project you're excited about and reach out to them directly. Most blockchain teams are willing to hire remote for the right talent. Many devs are readily accessible on Twitter, Github, or on their public Slack channels. If you have a solid portfolio and can demonstrate technical chops, most people will be impressed if you show some initiative.

But to my mind, the best way to get involved in a company is to find a project you're excited about and reach out to them directly. Most blockchain teams are willing to hire remote for the right talent. Many devs are readily accessible on Twitter, Github, or on their public Slack channels. If you have a solid portfolio and can demonstrate technical chops, most people will be impressed if you show some initiative.

And that's as far as I've got for you. If you've done all of the above, you should be set, and you'll probably be even farther along than me before long.

And that's as far as I've got for you. If you've done all of the above, you should be set, and you'll probably be even farther along than me before long.

The rabbit wormhole (The rabbit wormhole)

What I've shown you is just the beginning. Cryptocurrencies are still in their infancy, and I really believe the is the most rapidly evolving space you can be working in. I'm sure this guide will be out of date within a year, and there are so many amazing projects I haven't had the opportunity to talk about. But if you get into this space, you'll find them in due time.

What I've shown you is just the beginning. Cryptocurrencies are still in their infancy, and I really believe the is the most rapidly evolving space you can be working in. I'm sure this guide will be out of date within a year, and there are so many amazing projects I haven't had the opportunity to talk about. But if you get into this space, you'll find them in due time.

Keep exploring. Keep getting better. Keep learning.

Keep exploring. Keep getting better. Keep learning.

And I hope to see you come join us.

And I hope to see you come join us.

Haseeb

Haseeb

> 翻译自: https://www.freecodecamp.org/news/the-authoritative-guide-to-blockchain-development-855ab65b58bc/

区块链开发指南