

# Appendix G. Additional Educational Resources

*By Noah Gift*

We all know that one book, one course, or one degree isn't enough—in-  
stead, a process of continuous learning is the best method of staying up to  
date. One way to keep learning is to form a study group at your work,  
with your friends, or at school to keep growing. The following resources  
from classes I teach in ML engineering, MLOps, and applied computer vi-  
sion can help you start.

## Additional MLOps Critical Thinking Questions

- A former startup engineering manager mentions that “Agile” project management alone isn't enough to ship a minimum viable product (MVP). Often a 3-month weekly schedule is also needed (i.e., waterfall planning). Discuss your response to this opinion.
- What problems does a continuous integration (CI) system solve?
- Why is a CI system an essential part of SaaS software?
- Why are cloud platforms the ideal target for analytics applications?
- How does deep learning benefit from the cloud?
- What are the advantages of managed services like Google BigQuery?
- How does Google BigQuery differ from a traditional SQL?
- How does machine learning (ML) prediction directly from BigQuery add value to the Google platform?
- What advantages could this have for analytics application engineering?
- How does AutoML have a lower total cost of ownership (TCO)?
- How could it have a higher TCO?
- What problems do different environments solve?
- What problems do a different environment create?

- How can you properly manage unexpected costs in the cloud?
- What are three tools to help you manage costs on the Google Cloud Platform?
- What are three tools to help you manage costs on the AWS Platform?
- What are three tools to help you manage costs on the Azure Platform?
- Why can JavaScript Object Notation (JSON) logging often be better than unstructured logging?
- What are some downsides to alerts that go off too much?
- Create a life-long learning plan by answering the following questions: What skills are you going to learn this quarter, why, and how? What skills are you going to learn by the end of this year, why, and how? What skills are you going to learn by the end of next year, why, and how? What skills are you going to learn by the end of five years, why, and how?
- What problems does a continuous delivery (CD) system solve?
- Why is a continuous delivery system an essential part of data engineering?
- What is the critical difference between continuous integration and continuous delivery?
- Explain how monitoring and logging play a critical role in data engineering.
- Explain what can go wrong with health checks.
- Explain why “data governance” is the “unsung hero” of cybersecurity.
- Explain how testing plays a critical role in data engineering.
- Explain how automation and testing are so closely connected.
- Pick a favorite Python command-line framework and write a hello-world example, and share. Could you explain why you chose it?
- Explain how cloud computing is impacting data engineering.
- Explain how serverless is impacting data engineering.
- Share a simple Python AWS Lambda function and explain what it does.
- Explain what machine learning engineering is.
- Create and share a simple Dockerfile that runs a Flask app. Explain how it works.
- Explain what data engineering is.
- Truncate and shuffle a large dataset, load into Pandas, and share your work. Explain the approach you used.

- Explain what DevOps is and how it enhances a data engineering project.
- What problem does the cloud solve when considering real-world computer vision problems?
- How could Colab notebooks and Jupyter Notebooks be used to exchange ideas or build research portfolios?
- What are some critical differences between biological and machine vision?
- What are some real-world use cases of generative modeling?
- Explain how game-playing machines can use computer vision to win.
- What are the pros and cons of using computer vision APIs to solve real-world problems?
- How is AutoML impacting data science now, and how will it impact it in the future?
- What are the real-world use cases of edge-based machine learning?
- What are a few edge-based machine learning platforms?
- How do integrated platforms fit into existing companies' ML strategies?
- How could SageMaker change machine learning model creation in organizations?
- What are the practical use cases of AWS Lambda?
- Explain practical use cases of transfer learning. Please explain how you could use it in a project.
- How could you take an ML model you built to a more advanced phase of functionality?
- What is IAC, and what problem does it solve?
- How should a company decide on what level of cloud abstraction to use for a project: SaaS, PaaS, IaaS, MaaS, serverless?
- What are the different layers of network security on AWS, and what unique problems do each solve?
- What problem do AWS Spot instances solve, and how could you use them in your projects?
- Create a Docker format container and recommend how to use them in a project.
- Evaluate container management services like Kubernetes and hosted Kubernetes and create a solution with them.

- Summarize container registries and how to use them to create custom containers.
- What are containers?
- What problem do containers solve?
- What is the relationship between Kubernetes and containers?
- Accurately evaluate distributed computing challenges and opportunities and apply this knowledge to real-world projects.
- Summarize how eventual consistency plays a role in cloud native applications.
- How does the CAP Theorem play a role in designing for the cloud?
- What are the implications of Amdahl's law for machine learning projects?
- Recommend appropriate use cases for ASICS.
- Consider the implications of the end of Moore's Law.
- What are the problems with a "one size fits all" approach to relational databases?
- How could a service like Google BigQuery change the way you deal with data?
- What problem does a "serverless" database like Athena solve?
- What are the critical differences between block and object storage?
- What are the fundamental problems that a data lake solves?
- What are the trade-offs with a serverless architecture?
- What are the advantages of developing with Cloud9?
- What problems does Google App Engine solve?
- What problems does the Cloud Shell environment solve?

## Additional MLOps Educational Materials

Beyond the resources in this book, these are additional resources updated frequently that you can leverage to continue to improve:

- Many other MLOps-related videos are updated weekly on the O'Reilly Platform by [Pragmatic AI Solutions](#). You can also review the Katacodas on the O'Reilly Learning Platform.

- Several free books are available on the [Pragmatic AI Labs website](#).
- Subscribe to the [Pragmatic AI Labs YouTube Channel](#).
- Take the Duke and Coursera specialization [Building Cloud Computing Solutions at Scale](#).

## Education Disruption

A disruption is a break, interruption, or change from the established model or process. This section provides my thoughts on educational disruption and how it affects learning MLOps techniques.

Disruption is always easy to spot in hindsight. Consider the issue with taxi drivers and current services like Lyft and Uber. How did requiring drivers to pay [one million dollars for a taxi medallion](#) make sense as a mechanism to facilitate public taxi service ([Figure G-1](#))?



Figure G-1. Taxi medallion<sup>1</sup>

What were the problems that companies like [Lyft](#) and [Uber](#) solved?

- Lower price
- Push versus Pull (driver comes to you)
- Predictable service
- Habit-building feedback loop
- Async by design
- Digital versus analog
- Nonlinear workflow

Let's consider those same ideas in regard to education.

## Current State of Higher Education That Will Be Disrupted

A similar disruption is underway with education. Student debt is at an all-time high with a [linear growth rate from 2008, according to Experian](#) as shown in [Figure G-2](#).

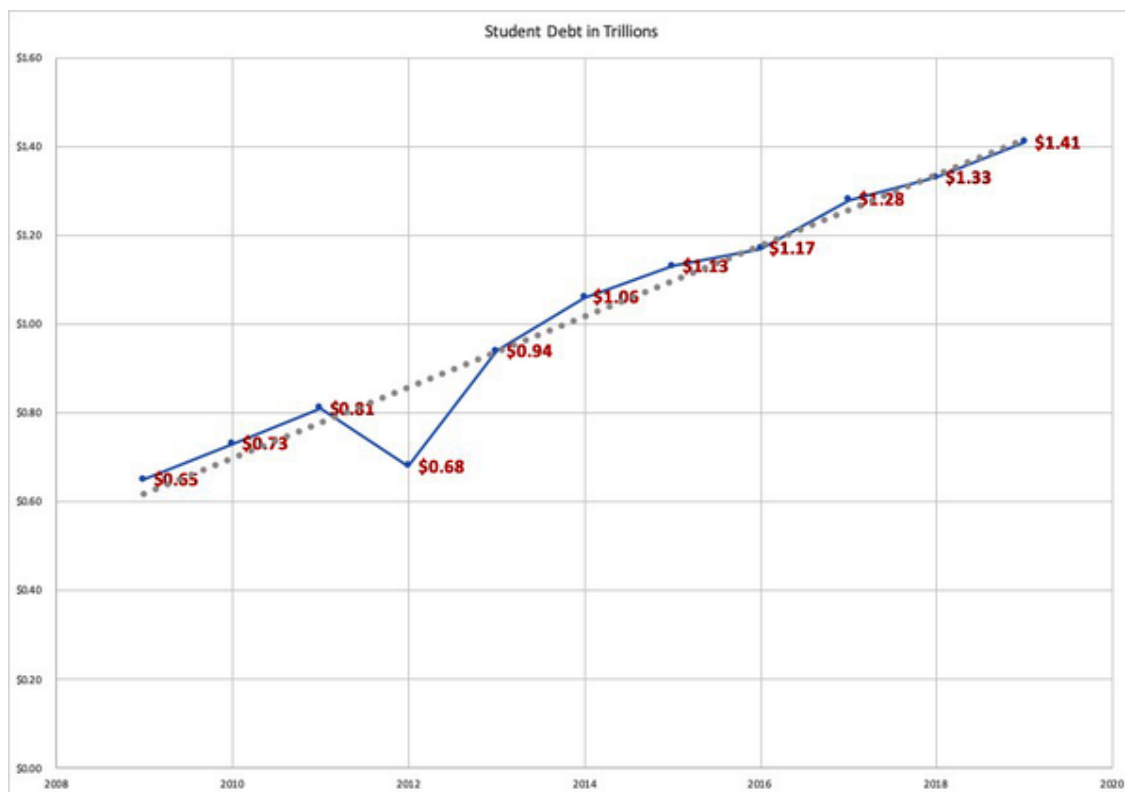


Figure G-2. Experian Student Debt

Simultaneous to that disturbing trend is an equally troubling statistic that [4/10 college grads in 2019 were in a job](#) that didn't require their degree ([Figure G-3](#)).



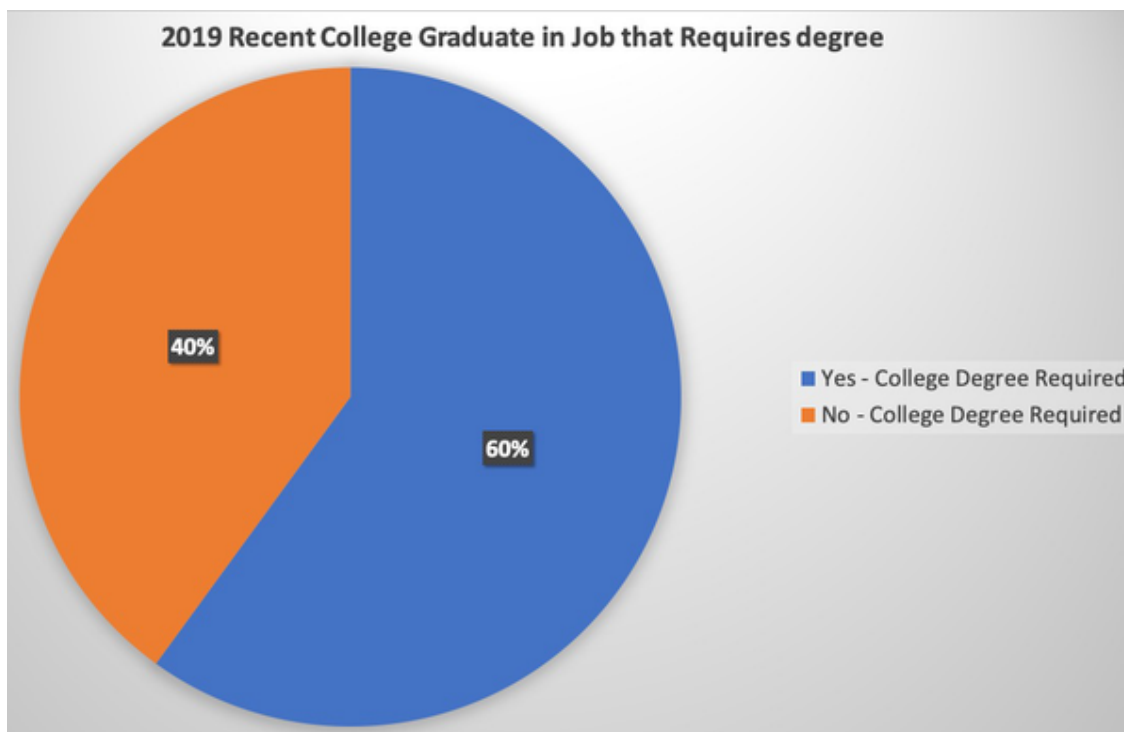


Figure G-3. Jobs requiring degree

This process is not sustainable. Student debt cannot continue to grow every year, and at the same time, produce almost half of the outcomes that do not lead directly to a job. Why shouldn't a student spend four years learning a hobby like personal fitness, sports, music, or culinary arts instead if the outcome is the same? At least in that situation, they would not be in debt and also have a fun hobby they could use for the rest of their life.

In the book *Zero to One* by Peter Thiel (Currency), he mentions the 10X rule. He states that a company will need to be 10X better than its closest competitor to succeed. Could a product or service be 10X better than traditional education? Yes, it could.

## 10X Better Education

So what would a 10X education system look like in practice?

### Built-in apprenticeship

If the focus of an educational program was jobs, then why not train on the job while in school?

## Focus on the customer

Much of the current higher-education system's focus is on faculty and faculty research. Who is paying for this? The customer, the student, is paying for this. An essential criterion for educators is publishing content in prestigious journals. There is only an indirect link to the customer.

Meanwhile, companies like [Udacity](#), Coursera, O'Reilly, and [Edx](#) are directly giving customers these goods. This training is job-specific and continuously updated at a pace much quicker than a traditional university.

It could be that skills taught to a student focus on getting a job outcome only. The majority of students are focused on getting jobs. They are less focused on becoming better human beings. There are other outlets for this goal.

## Lower time to completion

Does a degree need to take four years to complete? It may take that long if much of the time of the degree is on nonessential tasks. Why couldn't a degree take one year or two years?

## Lower cost

According to [USNews](#) the median four-year annual tuition is \$10,116 for a Public, In-State University; \$22,577 for a Public, Out-of-State University; and \$36,801 for a Private university. The total cost of getting a four-year degree (adjusted for inflation) has risen unbounded since 1985 ([Figure G-4](#)).



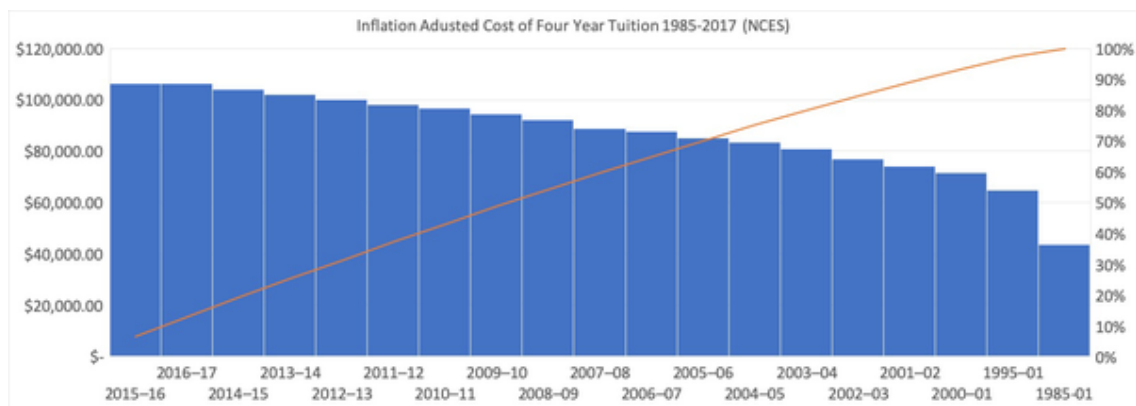


Figure G-4. Tuition inflation adjusted

Could a competitor offer a product that is 10X cheaper? A starting point would be to undo what happened from 1985 to 2019. If the product hasn't improved, but the cost has tripled, this is ripe for disruption.

## Async and remote first

Many software engineering companies have decided to become “remote first”. In other cases, companies like Twitter are moving to a distributed workforce. In building software, the output is a digital product. If the work is digital, the environment can be made entirely async and remote. The advantage of an async and remote first course is distribution at scale.

One of the advantages of a “remote first” environment is an organizational structure focused on the outcomes more than the location. There are tremendous disruptions and waste in many software companies due to unnecessary meetings, noisy working environments, and long commutes. Many students will be heading into “remote first” work environments, and it could be a significant advantage for them to learn the skills to succeed in these environments.

## Inclusion first versus exclusion first

Many universities publicly state how many students applied to their program and how few students were accepted. This exclusion-first-based approach is designed to increase demand. If the asset sold is physical, like a Malibu beach house, then yes, the price will adjust higher based on the market. If the product sold is digital and infinitely scalable, then exclusion doesn't make sense.

There is no free lunch, though, and [strictly boot camp style programs](#) are not without issues. In particular, curriculum quality and teaching quality shouldn't be an afterthought.

## **Nonlinear versus serial**

Before digital technology, many tasks were continuing operations. A good example is television editing technology. I worked as an editor for ABC Network in the 1990s. You needed physical tapes to edit. Soon the video-tapes became data on a hard drive, and that opened up many new editing techniques.

Likewise, with education, there is no reason for an enforced schedule to learn something. Async opens up the possibility of many new ways to learn. Mom could study at night; existing employees could learn on the weekend or during their lunch breaks.

## **Life-long learning: permanent access to content for alumni with continuous upskill path**

Another reason educational institutions should rethink going “remote first” is because it would allow for the creation of courses offered to alumni (for zero cost or a SaaS fee). SaaS could serve as a method of protection against the onslaught of competitors coming. Many industries require constant upskilling. A good example is the technology industry.

It would be safe to say that any technology worker needs to learn a new skill every six months. The current education product doesn't account for this. Why wouldn't alumni be given a chance to learn the material and gain certification on this material? Enhanced alumni could lead to an even better brand.

## **Regional job market that will be disrupted**

As a former Bay Area software engineer and homeowner, I don't see any future advantage to living in the region at the current cost structure ([Figure G-5](#)). The high cost of living in hypergrowth regions causes many

cascading issues: homelessness, increased commute times, dramatically lowered quality of life, and more.

### Homeownership Affordability Varies Across the Country - All Households

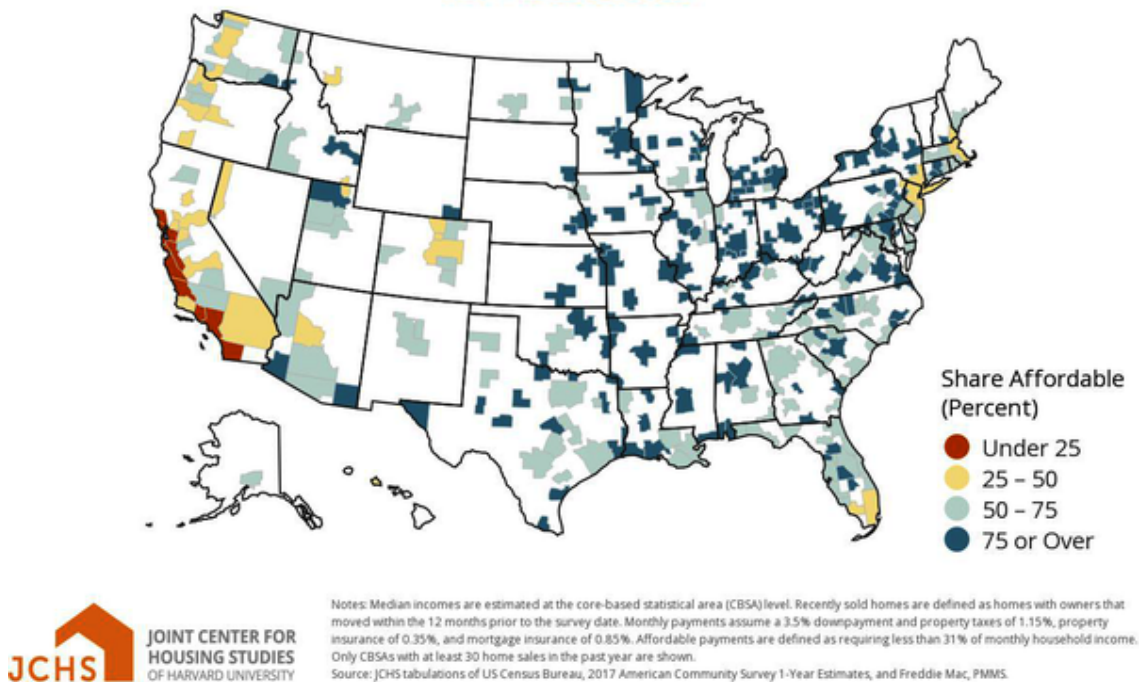


Figure G-5. Housing affordability in USA

Where there is a crisis, there is an opportunity. Many companies realize that whatever benefit lies in an ultra-high cost region, it isn't worth it. Instead, regional centers with excellent infrastructure and low cost of living have a massive opportunity. Some of the characteristics of regions that can act as job growth hubs include access to universities, transportation, low price of housing, and good government policies toward growth.

An excellent example of a region like this is Tennessee. It has [free associate degree programs](#) and access to many top universities, low cost of living, and has top research institutes like [Oak Ridge National Lab](#). These regions can dramatically disrupt the status quo, especially if they embrace remote first and asynchronous education and workforces.

### Disruption of hiring process

The hiring process in the United States is ready for disruption. It is easily disruptable because of a focus on direct, synchronous actions. [Figure G-6](#) shows how it would be possible to disrupt hiring by eliminating all inter-

views and replacing them with automatic recruitment of individuals with suitable certifications. This is a classic distributed programming problem that fixes a bottleneck by moving tasks from a serial and “buggy” workflow to a fully distributed workflow. Companies should “pull” from the workers instead of continually pulling on a resource locked up in futile work.

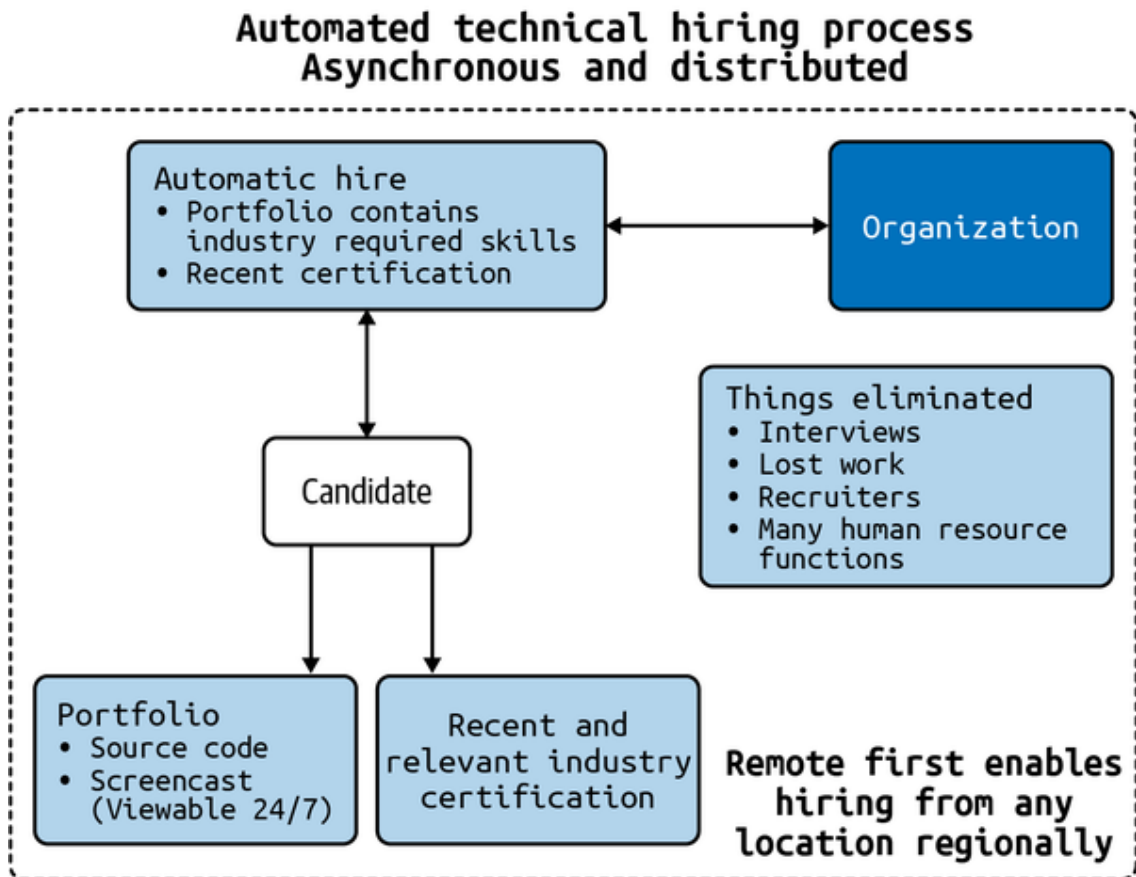


Figure G-6. Disrupt hiring

One reason why cloud computing is such an important skill is that it simplifies many aspects of software engineering. One of the issues it simplifies is building solutions. Many solutions often involve a YAML format file and a little bit of Python code.

## Conclusion

Education is no longer static. To be relevant in MLOps requires a growth mindset. This growth mindset means you must learn things on an ongoing basis while producing results. The good news is that the motivated

have an increasing chance to succeed if they take advantage of the explosion of technical training opportunities.

**1** Source: CALIFORNIA, SAN FRANCISCO 1996 -TAXI MEDALLION SUPPLEMENTAL LICENSEPLATE - Flickr - woody1778a.jpg.

[Support](#)   [Sign Out](#)

©2022 O'REILLY MEDIA, INC. [TERMS OF SERVICE](#) [PRIVACY POLICY](#)