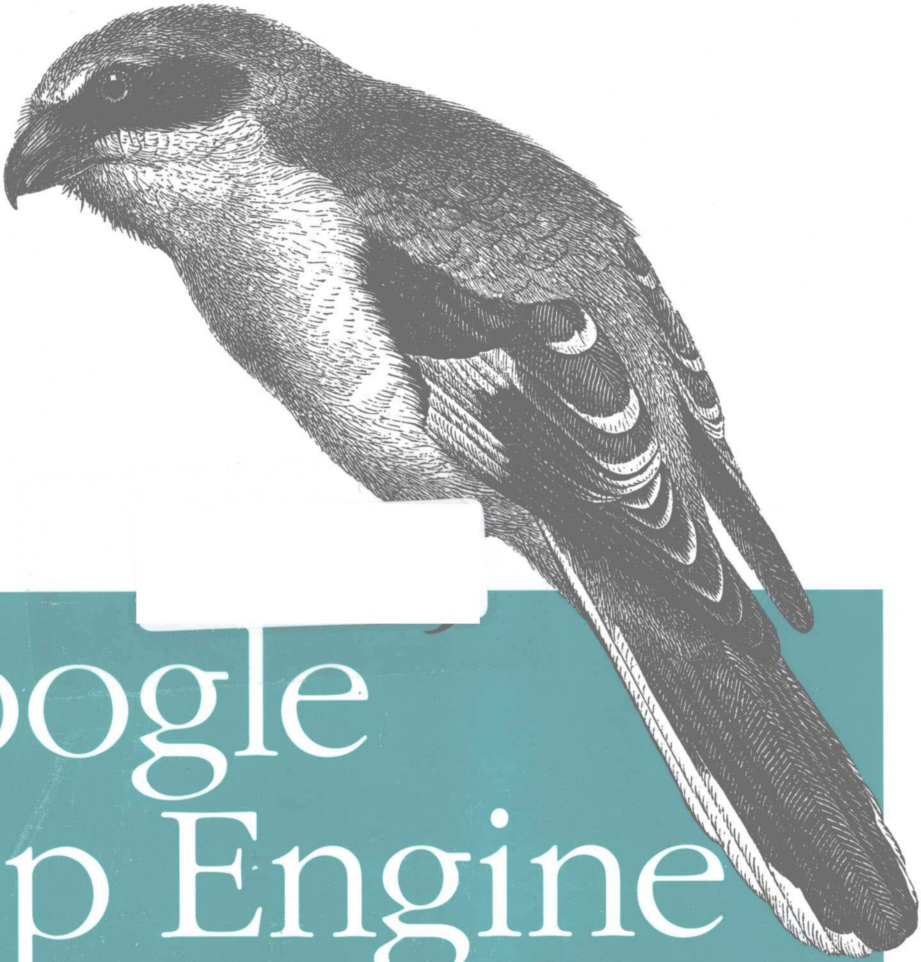
Google App Engine 开发（影印版）



Using Google App Engine

O'REILLY® Google™ Press

東南大學出版社

Charles Severance 著

Google App Engine 开发（影印版）

Using Google App Engine

Clarles Severance

O'REILLY®

Beijing • Cambridge • Farnham • Köln • Sebastopol • Taipei • Tokyo

O'Reilly Media, Inc. 授权东南大学出版社出版

东南大学出版社

图书在版编目（CIP）数据

Google App Engine 开发：英文/（美）斯华伦（Severance, C.）著.—影印本.—南京：东南大学出版社，2010.6

书名原文：Using Google APP Engine
ISBN 978-7-5641-2268-3

Ⅰ.①G… Ⅱ.①斯… Ⅲ.①主页制作－程序设计－英文 Ⅳ.①TP393.092

中国版本图书馆 CIP 数据核字（2010）第 089118 号

江苏省版权局著作权合同登记
图字：10-2010-156 号

Google App Engine 开发（影印版）

出版发行：东南大学出版社
地　　址：南京四牌楼 2 号
出 版 人：江　汉
网　　址：http://press.seu.edu.cn
电子邮件：press@seu.edu.cn
印　　刷：扬中市印刷有限公司
开　　本：787 毫米 × 980 毫米　16 开本
印　　张：16.5 印张
字　　数：323 千字
版　　次：2010 年 6 月第 1 版
印　　次：2010 年 6 月第 1 次印刷
书　　号：ISBN 978-7-5641-2268-3
印　　数：1~1800 册
定　　价：52.00 元（册）

本社图书若有印装质量问题，请直接与读者服务部联系。电话（传真）：025-83792328

Preface

The greatest single reason that the World Wide Web has been so widely used and adopted is because individuals are allowed to participate in the Web. People can produce web content and create a MySpace page or home pages provided by their school or organization and contribute their creativity and content to the Web. Free services like Blogger, Flickr, Google Sites, Google Groups, and others have given us all an outlet for our creativity and presence on the Web—at no charge.

For most of the life of the Web, if you wanted to have your own rich software-backed website with data storage, your only choice was to purchase hosting services from an Internet Service Provider (ISP) and learn database management and a programming language like PHP to build or run your software. Learning and paying for this much technology was just beyond the reach of most web users, who simply had to accept the limited features of MySpace, Blogger, or whatever system hosted their web content.

In April 2008, Google announced a product called App Engine. When you write a program for the Web that runs on App Engine, your software runs on the Google servers somewhere in the Google "cloud." It is as if you are a Google employee and you have access to the entire scalable Google infrastructure. App Engine captures much of Google's experience of building fast, reliable, and scalable websites, and through App Engine, Google is revealing many of the secrets about how its own applications scale to millions of users.

The most exciting part of the Google App Engine announcement is the fact that it is free for moderate levels of use. Every person with a Gmail account can have a number of free applications running on the Google infrastructure. If your application becomes extremely popular and your traffic goes above the allowed levels of the free account, you can pay to use more of Google's resources. As your application scales, Google engineers and operations staff take care of all the hardware, data storage, backup, and network provisioning for you.

The cost of purchasing resources from Google's cloud of servers is likely far less than purchasing/renting/maintaining the same amount of resources on your own. Google focuses on providing hardware and network; you focus on building your application and the user community around your application.

Maybe you could write the next Twitter, Craigslist, or del.icio.us. Maybe your idea will be the next big thing that will take off and you can "retire" on the revenue from Google AdWords. Or maybe you just want a site for your local off-road motorcycle club to publish its newsletter, share crash pictures, and maintain a mailing list.

Google App Engine removes the cost barrier from building and deploying software and data-backed websites and putting those sites into production. This book aims to make it easier for the average user to build and deploy basic websites using Google App Engine.

The hope is that literally millions of people from around the world will now be empowered to program on the Web. Who knows what creative applications will evolve in this new and exciting era?

Who Should Read This Book?

This book is aimed at anyone who wants to get started with Google App Engine. Perhaps you are a seasoned programmer with many other languages under your belt; perhaps you have played a bit with HTML and CSS, and you want to learn about software and data-backed websites by deploying your own site or application. It's written for anyone who wants to learn about this new and exciting capability previously reserved for the technical elite.

The book assumes no existing knowledge of programming or web technologies and is written in a way that is understandable to nonprogrammers. It starts from the beginning and covers all the necessary prerequisite technologies, including the Python programming language, HyperText Markup Language (HTML), Cascading Style Sheets (CSS), and the HyperText Transport Protocol (HTTP).

In fact, this book's secret plan is to transform someone from with no knowledge about web technologies into a fire-breathing web application developer in less than a week. By the end of this book, you will know at least enough about these web technologies to be dangerous to yourself and to others. You will have built and understood a fully working and deployed Google App Engine program that touches on all the major technical aspects of the App Engine environment, and you will be in an ideal position to extend your knowledge using Google's online materials or other books to dig more deeply into the subject.

What's in This Book?

This book uses a consistent example of a website with members and multiuser chat, which is built continuously throughout the book. The example is used to introduce topics from HTML and CSS all the way through using AJAX to update your pages dynamically without redrawing the entire screen.

Although I'll cover a lot of material, coverage is limited to include only the information that you need to know to build your application. Once you venture into building more sophisticated applications, you will need additional books and online resources on HTML, CSS, Python, jQuery, and JavaScript.

Chapters 1 through 4 cover the necessary background material in the web technologies that are brought together in the book. If you have experience with any of the topics in Chapters 1 through 4, you can safely skip those chapters (but they'll still be there in case you have a question or need a refresher).

Chapter 1, *Programming on the Web*
: Programming in Google's production environment is different from running your own server or using a hosting account on an ISP. Google takes care of everything related to running your application in production. The trade-off is that you need to follow Google's rules and be a good citizen in Google's community of other applications. This chapter provides a description of the cloud and how it is different from being responsible for your own servers, plus it helps to explain some of the nuances of the App Engine environment.

Chapter 2, *HTML and CSS*
: I assume that folks know the basics of HTML, but there are some important bits that must be covered so that your pages are nice and clean. In the last few years, the legacy browsers that did not support modern HTML and CSS have pretty much died out, so we can write simple and clean HTML and leave the formatting to CSS. I also explore how to validate your HTML and CSS and conform to the document type (DOCTYPE). I talk about page layout using CSS and introduce a bit of the CSS block model so that you can make pretty web pages with simple navigation. If you have been learning HTML by viewing the source code of other people's MySpace pages, you probably need a refresher on the "modern" way to design pages using HTML and CSS.

Chapter 3, *Python*
: This is a very quick introduction to Python that covers only the areas of Python that are necessary for reading the rest of the book. Because we are writing a web application and not a general-purpose application, you need to know only a subset of Python. Python is a great language for beginners, casual users, and power users because it is both simple and powerful. Many claim that Python is *the* language for people who actually use computers.

Chapter 4, *Sending Data to Your Application*
: This chapter sounds a little nerdy—and it is! I think that you actually need to know how the browser talks to a web server and exchanges data using HTTP. It is not all that complex, once you understand it—and it's worth learning. This chapter introduces the first simple App Engine program that we will use to explore how the HTTP request/response cycle works from both the browser and server perspectives.

Chapter 5, *The App Engine webapp Framework*

Properly written App Engine programs consist of a set of cooperating objects. The object-oriented design pattern is how we create and link these objects to get our work done. In this chapter, I teach the basics of object-oriented Python and then jump right into a sample App Engine program using the Google object-oriented web framework. Like the rest of the background chapters, I explain the basics of objects in Python by covering only what you need to know for App Engine.

Chapter 6, *Templates*

In this chapter, I introduce the first part of the Model-View-Controller pattern used in most web frameworks. Using templates, I separate the look and feel of the application (the View) from the rest of the logic of the application. Templates are files that contain HTML augmented using the Django template language to allow certain areas of the HTML to contain information that comes from your Python code (the Controller). You will learn about basic templates as well as inherited templates—where common material is kept in one file and reused across many files—object-oriented templates, as it were.

Chapter 7, *Cookies and Sessions*

In this chapter, I introduce the concept of a session. Sessions and cookies combine to allow the web server to work with multiple simultaneous users. Sessions associate bits of information, such as the name of the currently logged-in user, with one particular browser so that it can distinguish which incoming requests come from which browser.

Chapter 8, *App Engine Datastore*

Google App Engine does not provide you with a relational database. Experts in relational databases will likely feel a bit out of their element when they first look at the Google App Engine Models and Datastore. Readers who have never learned relational databases can be quite thankful that Models (as in Model-View-Controller) are much simpler to use than relational databases. Also, Google has learned through experience that relational databases simply cannot scale to levels beyond millions of users. The Google Datastore can be scaled well beyond a million users. Although you may never need to scale to several million users, you will like how using Models makes storage easier.

Chapter 9, *JavaScript, jQuery, and AJAX*

This chapter adds a little in-browser interactivity to our application via jQuery and AJAX to implement a simple multiuser chat. It also covers how you create multiple data models and link data objects together in the Google Datastore. I explain just enough JavaScript, jQuery, and AJAX to help you understand how your application works with these technologies.

Chapter 10, *Running Your Application on the Google Infrastructure*

This chapter covers how to run your application in the Google infrastructure cloud. You will learn how to get your free App Engine accounts and then upload your software into the cloud. You also learn about the administration interfaces that allow you to monitor and manage your application and data while you are in production.

Chapter 11, *Memory Cache*
: The App Engine Memory Cache is a critical technology for making fast and scalable websites. Clever use of Memory Cache can dramatically reduce the load on a Datastore or the network and increase application responsiveness, particularly for material that is read over and over. In this chapter, we explore how the Memory Cache works and develop simple Session capability using the Memory Cache.

Teaching with This Book

This book came out of a University of Michigan School of Information course titled "Design of Complex Websites (SI539)." This course explores emerging web technologies, including ColdFusion, PHP, Ruby on Rails, and now Google Application Engine. The basic idea of the course was to teach in one semester students with very limited technical background enough about database-backed web development "to be dangerous to themselves and others." The course and book are aimed at introducing these concepts to a nontechnical audience.

The book is written at a beginning level; I think that it can be used to support a semester-long "Introduction to Web Programming" course from high school through graduate school. Because this book includes introductions to Python and to HTML and CSS, I hope that it can be used by itself or with supporting material.

For beginning students, you can have a series of assignments that are very similar to the examples in the book, with minor changes such as color or adding images to pages. The assignments can be made more difficult by having the students do a series of parallel, but different, projects that correspond roughly to the concepts in the book's running examples.

The book can also be used to support a one-day workshop in App Engine. It would probably be difficult to teach Python, HTML, CSS, and App Engine in a single day. But because the examples are a single evolving application and each example builds on the previous one, it is possible to skip steps in the interest of time. You might have one exercise where the students modify the *ae-08-login* example (login without session) to produce *ae-09-session* (login with session) and then skip ahead to modify the *ae-11-chat* (non-AJAX chat) to produce *ae-12-ajax* (AJAX-based chat). The chapters walk readers through the necessary changes from each version of the application to the next.

To help support the use of the book in a classroom setting, I provide freely reusable classroom materials that make it easier to use the book in other courses at my personal website (*http://www.dr-chuck.com*). I would love to hear from other teachers who use the book so that we can all share our experiences, assignments, tips, and lecture materials.

Conventions Used in This Book

The following typographical conventions are used in this book:

Italic

Indicates new terms, URLs, email addresses, filenames, file extensions, pathnames, directories, and Unix utilities.

`Constant width`

Indicates commands, options, switches, variables, attributes, keys, functions, types, classes, namespaces, methods, modules, properties, parameters, values, objects, events, event handlers, XML tags, HTML tags, macros, the contents of files, or the output from commands.

`Constant width bold`

Shows commands or other text that should be typed literally by the user.

`Constant width italic`

Shows text that should be replaced with user-supplied values.

This icon signifies a tip, suggestion, or general note.

This icon indicates a warning or caution.

Using Code Examples

This book is here to help you get your job done. In general, you may use the code in this book in your programs and documentation. You do not need to contact us for permission unless you're reproducing a significant portion of the code. For example, writing a program that uses several chunks of code from this book does not require permission. Selling or distributing a CD-ROM of examples from O'Reilly books *does* require permission. Answering a question by citing this book and quoting example code does not require permission. Incorporating a significant amount of example code from this book into your product's documentation *does* require permission.

We appreciate, but do not require, attribution. An attribution usually includes the title, author, publisher, and ISBN. For example: "*Using Google App Engine*, by Charles Severance. Copyright 2009 Charles Severance, 978-0-596-80069-7."

If you feel that your use of code examples falls outside fair use or the permission given above, feel free to contact us at *permissions@oreilly.com*.

Safari® Books Online

When you see a Safari® Books Online icon on the cover of your favorite technology book, that means the book is available online through the O'Reilly Network Safari Bookshelf.

Safari offers a solution that's better than e-books. It's a virtual library that lets you easily search thousands of top tech books, cut and paste code samples, download chapters, and find quick answers when you need the most accurate, current information. Try it for free at *http://my.safaribooksonline.com*.

Comments and Questions

Please address comments and questions concerning this book to the publisher:

O'Reilly Media, Inc.
1005 Gravenstein Highway North
Sebastopol, CA 95472
800-998-9938 (in the United States or Canada)
707-829-0515 (international or local)
707-829-0104 (fax)

We have a web page for this book, where we list errata, examples, and any additional information. You can access this page at:

http://www.oreilly.com/catalog/9780596800697

To comment or ask technical questions about this book, send email to:

bookquestions@oreilly.com

For more information about our books, conferences, Resource Centers, and the O'Reilly Network, see our website at:

http://www.oreilly.com

Acknowledgments

Writing a book is always an engrossing experience for me. It tends to take over my life and spare time and consumes all my free energy until the book is done. By the time the book is completed, it is amazing how many other people have had a significant contribution to the quality of the resulting product. In a sense, although one person's name is on the front cover, this is truly the work of a wise crowd of great friends who have given me so much help and support.

This is my second book with Mike Loukides as my editor, and once again, it was a joy to work with him. Mike is so good at working with an author who is also a busy academic and is trying to juggle classes, travel, consulting, and research along with writing.

For this book, Judy Loukides was also a great help in getting the book together on time. Judy jumped in and helped at a very crucial moment when time was running out and her help is greatly appreciated.

I have two good friends, mentors, colleagues, and coauthors in Noah Botimer and Gonzalo Silverio. Gonzalo has always been my guide to learning CSS throughout the Sakai project and as I taught CSS in my courses at the University of Michigan. He taught me that CSS was really clean and straightforward. It is particularly straightforward if you can always run to Gonzalo when you run into a difficult bit. Noah has always been there to help me figure out the really difficult things. He is great at digging into how something really works and helping me understand it well enough to teach the material to my students and take all the credit.

The technical reviewers did a great job of making sure that the book was sound. Trek Glowaki, Nick Johnson, Steven Githens, Kirby Urner, and Matt Simmons all did a great job in a very short time frame. I also want to thank Pete Koomen of Google for his encouragement at the 2008 Google I/O conference and throughout the process.

Paul Resnick, Sean Munson, Jim Eng, Marc Alier, and Jordi Piguillem Poch took the risk of using the book in their courses even before it was published. I very much appreciate their feedback and guidance as well as the feedback I got from their students. I need to thank the students from the "Design of Complex Websites" course at the University of Michigan in Fall 2008, who showed amazing patience as I gave them the earliest versions of each chapter, often produced only a few hours before lecture. They read the chapters carefully, patiently pointed out places where the narrative "dropped the ball," and reread the revised versions of the chapters.

I certainly appreciate how my students, friends, and colleagues gave me the space and time to write the book.

And I want to thank my parents, Marcia and Russ, for setting high expectations and helping me to learn to always think outside the box and always be prepared for new experiences. I want to thank my wife, Teresa, and my children, Amanda and Brent, for being patient during those months where I pretty much had no time for anything else except "the book."

Table of Contents

CHAPTER 1

Programming on the Web

You probably have been using the Web now for many years to read news, shop, gather information, and communicate with your friends. You start your web browser (Internet Explorer, Firefox, Safari, Opera, Chrome, or another) and navigate around the Web. You may even have a MySpace page or a blog somewhere and have written a bit of HyperText Markup Language (HTML) to customize the look of your page. Some web pages are just flat content, where you see the exact same thing every time you visit that page, and other pages have highly dynamic content and display something very different based on your actions or what you type.

In this book, you will learn how to write applications that generate those dynamic web pages and how to run your applications on the Google App Engine infrastructure.

A perfect example of an interactive and dynamic page is Google Search (Figure 1-1).

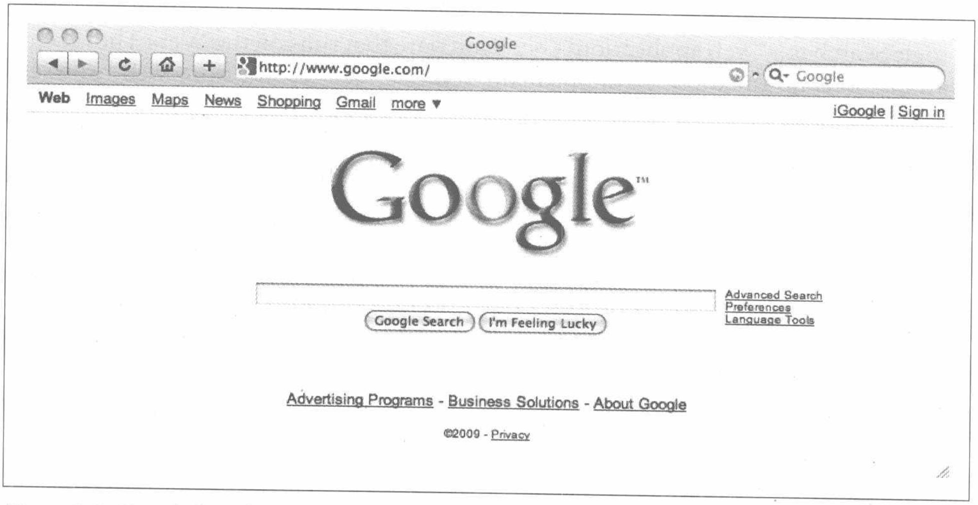


Figure 1-1. Google Search