

Programming in Go: A Beginner's Mind

How a complete beginner picked up Go as her first
backend language in 5 weeks



Audrey Lim
audreylh@gmail.com

Practicing lawyer
A curious cat
I love math, literature,
photography, beauty in the
abstract
Dad ex-programmer

How I became interested in programming

How does the web work?

How do computers work?

April 2014

Front-end programming

HTML CSS JS

End June 2014

Python?

Django

PIP

Virtualenv

South

Python 2.7 or 3.3?

MVC

Ruby?

Rails

RVM

MVC

ORM

Node?

Express

Roadblocks

- Development environment setup
- Installation problems
- Frameworks
- MVC architecture
- Too much information I don't need just to get started on web fundamentals

End June 2014

“Why don’t you try Go”

Ruby on Rails: Too much magic for a beginner

Python: You don’t learn this to build websites!

Node.js: Javascript is not for backend!

Go: It is too difficult for a beginner!

It was designed for advanced programmers!

Go

It's just Go 1.3

1. Easy installation
2. No reliance on frameworks / MVC
3. Go's libraries alone seemed a good exposure to CS concepts
4. Simplicity, liberty, focus

15 July 2014

Let's go with Go

Go as an excellent
teaching tool

1. Syntax
2. Mini challenges
3. Projects

- **The Way to Go** - Ivo Balbaert
- **An Introduction to Programming in Go** - Caleb Doxsey
- **How Do I Write And Deploy Simple Web Apps With Go?** - Satish Talim
- <http://tour.golang.org/>
- <http://gobyexample.com>

1. CRUD app
2. API Client
3. Raspberry Pi app
4. Micro-controller library
5. Markdown Parser

Micro-controller library

vspark: Go wrapper around C
program to control the Spark Core

C program: VoodooSpark
Javascript library

<https://github.com/audreylim/vspark>

Markdown Parser

Marshmallow: Simple markdown processor that converts markdown text into HTML

<https://github.com/audreylim/marshmallow>

\$ marshmallow file.md

Hello Denver

<h1>Hello Denver</h1>

Hello Denver

Nice to see so many **Gophers** today



<p>Nice to see so many Gophers <i>today</i></p>

Nice to see so many **Gophers** today

* Awesome
* Great

Awesome
Great

- Awesome
- Great

file.md

file.html

Markdown parser

Ben Johnson's Handwritten Parsers & Lexers in Go:

<http://blog.gopheracademy.com/advent-2014/parsers-lexers/>

Rob Pike's Lexical Scanning in Go: [https://](https://www.youtube.com/watch?v=HxaD_trXwRE)

www.youtube.com/watch?v=HxaD_trXwRE

Markdown parser

Concepts:

Parsing algorithms,
Context-free grammar, LR
parser

Finite state machines

Abstract syntax tree

```
type stateFn func(*Parser) stateFn

// Start state machine.
func (p *Parser) Parse() {
    p.run(p.stateParse())
}

// State machine: Chain states until EOF.
func (p *Parser) run(state stateFn) stateFn {
    newstate := state
    if newstate != nil {
        p.run(newstate)
    }
    return nil
}
```

Built in 2 weeks

Go projects and learning
were entirely outside my
day job

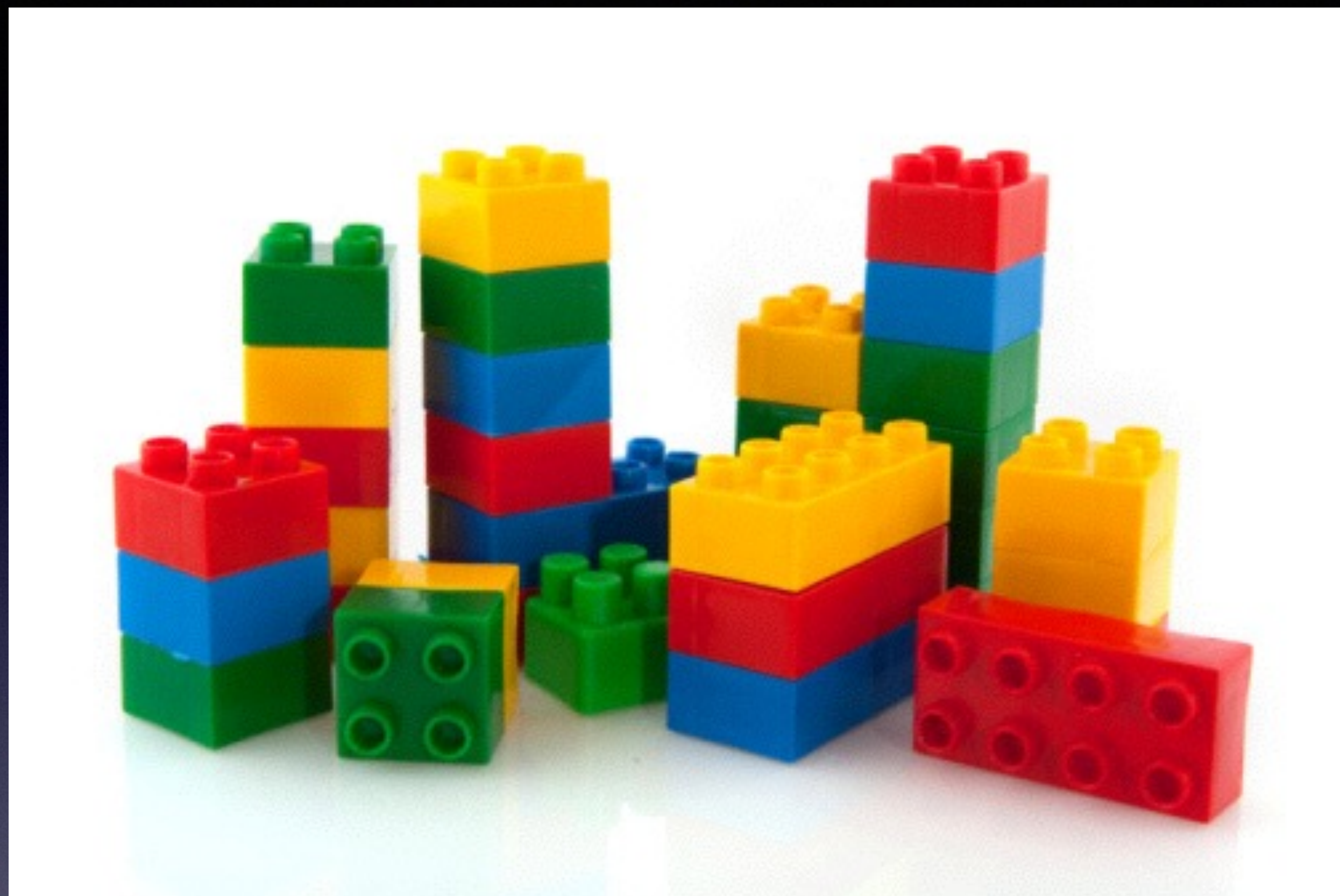
Timeline

April 2014	HTML, CSS, JS
July 2014	Learned Go
August 2014	CRUD app
August 2014	API client
September 2014	Raspberry Pi app
December 2014	Micro-controller library
February 2015	Markdown parser
March 2015	Grep utility in Rust
	Joined Nitrous.IO

Go isn't that difficult for beginners.
In fact, it is pretty awesome as a beginner language.

Go as a tool for the imagination

For me I found that the kind of programming tool you use or learn greatly shapes the way you think and the character of the programs you end up building.



Lego Bricks

Go's libraries, functions, packages

Freedom of imagination

Like building a house with lego bricks

Single Rule

1. Clear conventions
2. Usually only one way to do one thing
3. Without worrying too much about syntax issues, frees up your imagination to build whatever you want with very powerful tools

Go-Globe

<http://secure-falls-6445.herokuapp.com/>

Go-Globe

1. Pure Go
2. Libraries used

"encoding/json"

"fmt"

"html/template"

"io/ioutil"

"log"

"math/rand"

"net/http"

"os"

"strconv"

"time"

“The big idea is “**messaging**” ... The key in making great and growable systems is much more to design how its modules **communicate** rather than what their internal properties and behaviors should be.”

– **Alan Kay**

Why I love Go

Many times developers came to me and pointed out all the things that they didn't like about Go.

I think what's more important, for a beginner, is that with Go in a very practical way I was able to pick up programming / CS concepts rather quickly, enjoy the process, and learn a ton of things in the meantime.

Why I love Go

Go not only sustained but fuelled the curiosity about programming I already had inside.

Besides the technical learning, Go has also had a very direct and real impact on my life. I love what I do - programming. I found that “something else” that I’ve been searching for while I was a law student / lawyer.

Why I love Go

As I continue in my journey with programming,
I hope to retain this spirit of discovery, a
beginner's mind.

A beginner's mind of curiosity, amazement,
and wonder at what cumulative efforts toward
the progress of technology can do, and what
you can do with technology.

“Go will make you love
programming ~~again~~”

—experienced Go programmers

“Go made me love programming.”

Thank you.

Audrey Lim

Developer, Nitrous.IO

audreylh@gmail.com

@audreylim77

github.com/audreylim