# THE SUBTLE ART OF GIVING CODE WALKTHROUGH TALKS

RICARDO FERREIRA

PRINCIPAL DEVELOPER ADVOCATE
ELASTIC

@RIFERREI

# WHY WORRY ABOUT LIVE DEMOS?

elastic

elastic

# WHAT?

- BEHAVIORAL ASPECTS OF LIVE DEMOS
- FIXING CLASSIC LIVE DEMO ~~MISTAKES~~
- KEEPING THE ATTENTION SPAN HIGH

NOT ABOUT YOU.

IT IS ABOUT THEM.

elastic

YOU'RE MY BOY, BLUE
BLUE

# SHOW THAT YOU ENJOY WHAT YOU'RE DOING



elastic

@RIFERREI

elastic

# PAUSING IS KEY



elastic

elastic

# START WITH A REPO



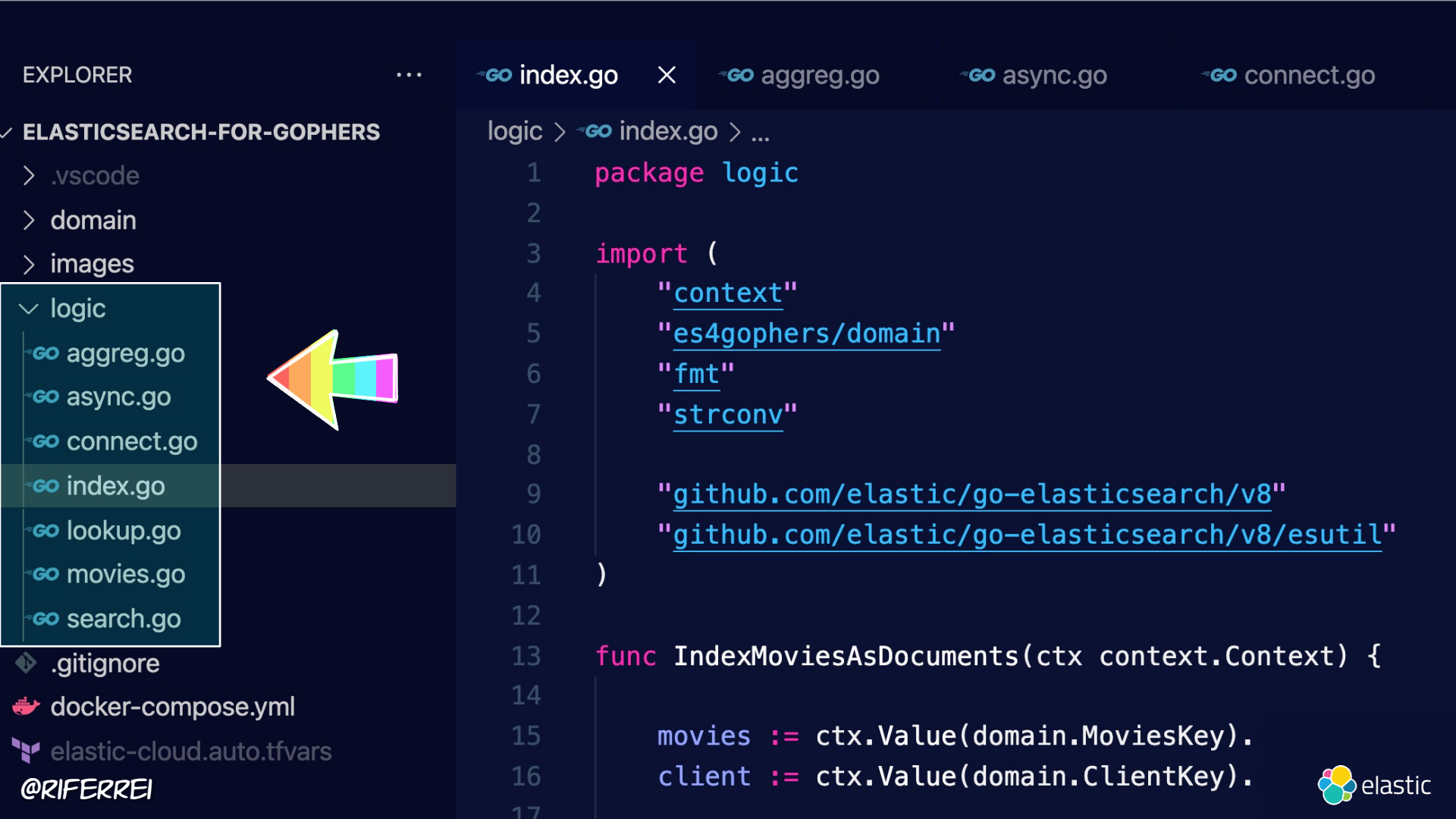
elastic

elastic

# BREAK YOUR CODE INTO A NARRATIVE



elastic


EXPLORER
index.go
aggreg.go
async.go
connect.go
ELASTICSEARCH-FOR-GOPHERS
.vscode
domain
images
logic
aggreg.go
async.go
connect.go
index.go
lookup.go
movies.go
search.go
.gitignore
docker-compose.yml
elastic-cloud.auto.tfvars
logic > index.go > ...
1 package logic
2
3 import (
4 "context"
5 "es4gophers/domain"
6 "fmt"
7 "strconv"
8
9 "github.com/elastic/go-elasticsearch/v8"
10 "github.com/elastic/go-elasticsearch/v8/esutil"
11 )
12
13 func IndexMoviesAsDocuments(ctx context.Context) {
14
15 movies := ctx.Value(domain.MoviesKey).
16 client := ctx.Value(domain.ClientKey).
@RIFERREI
elastic

# CREATE A PARTIALLY FINISHED BRANCH



elastic

GIT DIFF LIVESTREAM .. MAIN

MISTAKES
ARE FINE.

I MEAN IT.

elastic

# EASY ON YOUR PROGRAMMING LANGUAGE GIMMICKS



elastic

"THIS READS THE MOVIES FILE"

```
for i := 0; i < concurrency; i++ {
    waitGroup.Add(1)
    go func(workQueue chan string, waitGroup *sync.WaitGroup) {
        for entry := range workQueue {
            movieRaw := domain.MovieRaw{}
            err := json.Unmarshal([]byte(entry), &movieRaw)
            if err == nil {
                movie := func(movieRaw domain.MovieRaw) domain.Movie {
                    return domain.Movie{
                        Title:       movieRaw.Title,
                        Year:        movieRaw.Year,
                        RunningTime: movieRaw.Info.RunningTime,
                        ReleaseDate: movieRaw.Info.ReleaseDate,
                        Rating:      movieRaw.Info.Rating,
                        Genres:      movieRaw.Info.Genres,
                        Actors:      movieRaw.Info.Actors,
                        Directors:   movieRaw.Info.Directors,
                    }
                }(movieRaw)
                mutex.Lock()
                movies = append(movies, movie)
                mutex.Unlock()
            }
        }
        waitGroup.Done()
    }(workQueue, waitGroup)
}
```

# DON'T BE ALL ABOUT YOUR FAVORITE IDE

elastic

# KEEP THE TECH LIST TO A MINIMUM



elastic

"I JUST WANT DINNER 😰"

# EXPLAIN THE NON-OBVIOUS DETAILS

➡️ KUBECTL APPLY -F
@RIFERREI
elastic

# PROVIDE RESOURCES, NOT CALL-TO-ACTION

@RIFERREI

elastic

# THANK YOU

@RIFERREI

elastic