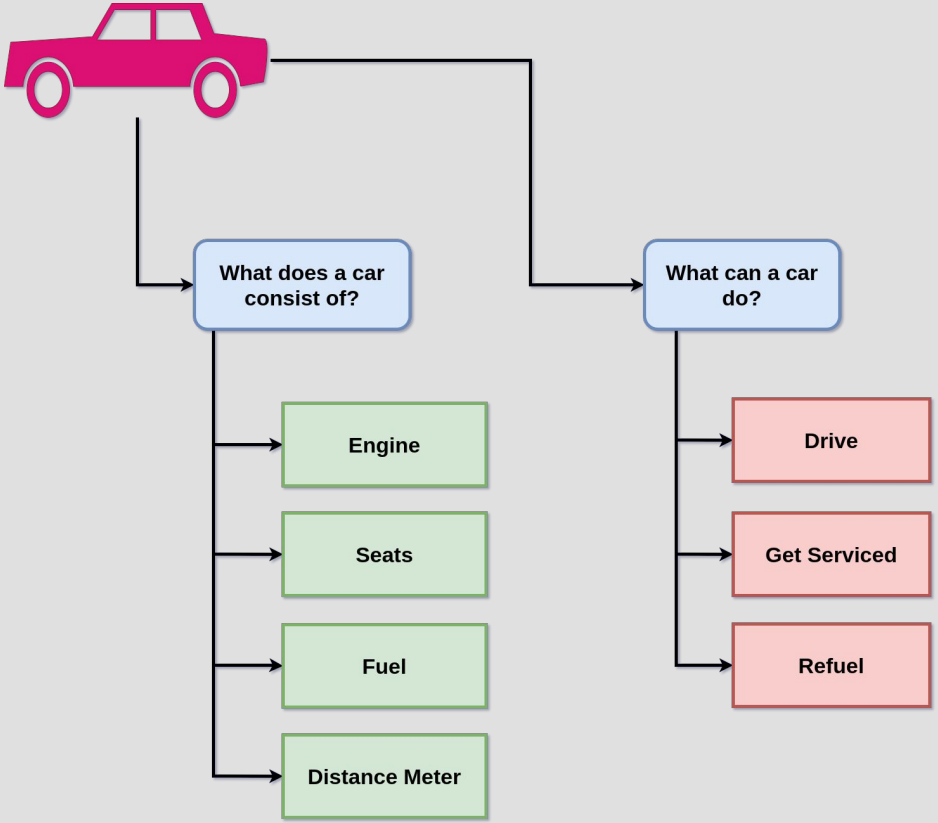
Getting Started with Google GoLang

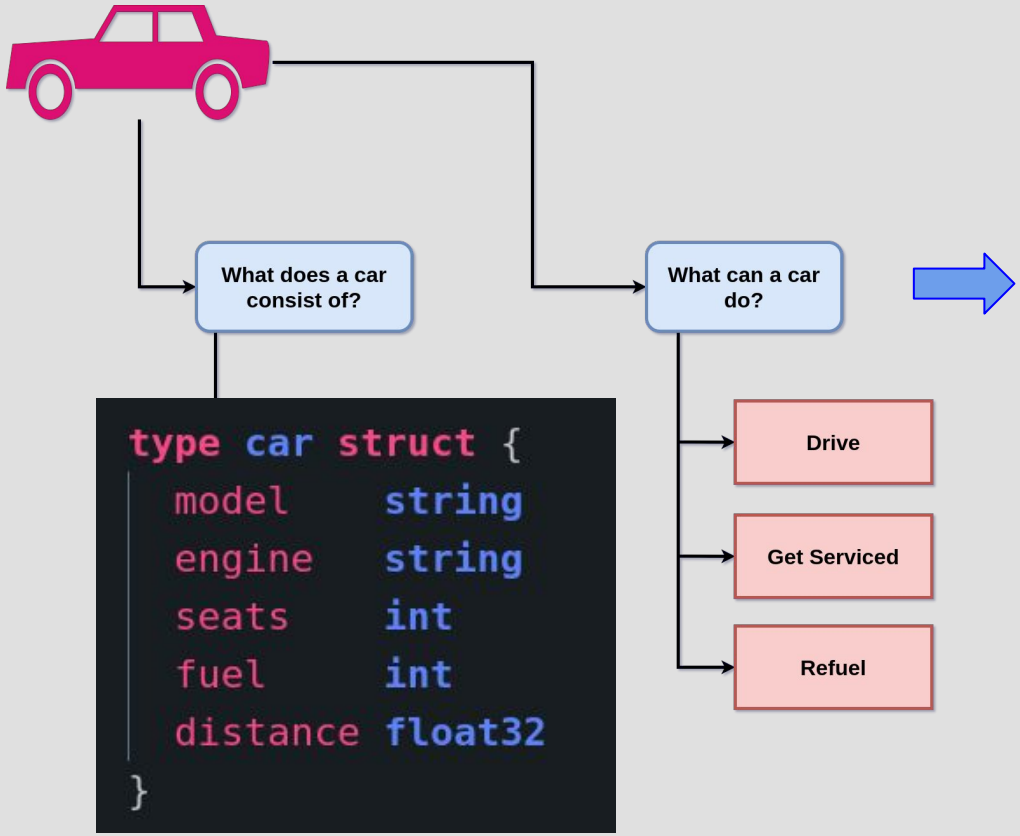
...

Week 2 - Receivers and Interfaces

Custom Types



How to represent this in code?

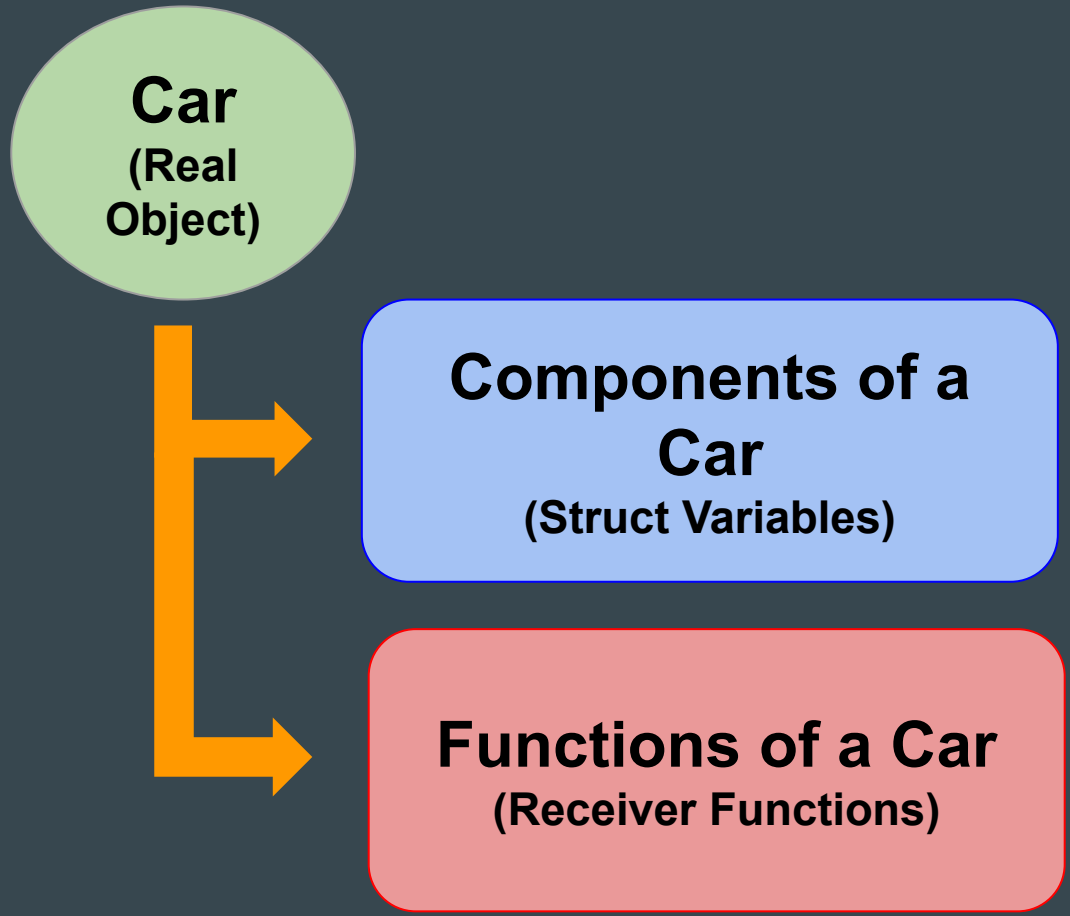



```go
type car struct {
    model    string
    engine   string
    seats    int
    fuel     int
    distance float32
}
```

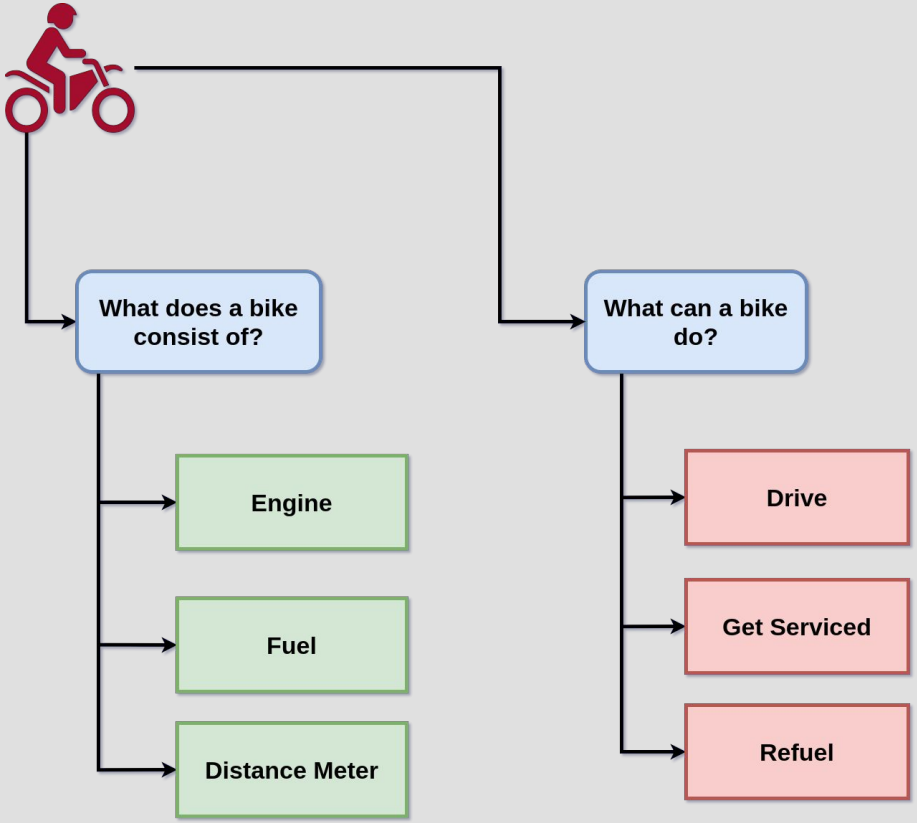
```go
func (c car) drive() {
    for c.fuel > 0 {
        c.fuel -= 1
        c.distance += 12.5
    }
}

func (c car) service() int {
    if c.engine == "v8" {
        return 2000
    } else {
        return 1000
    }
}

func (c car) refuel(fuel int) {
    c.fuel += fuel
}
```

The Analogy

- Custom types represent real world objects.
- Receivers (methods) define things the type (object) can do.



Another Custom Type



Bike, in code

What does a bike consist of?

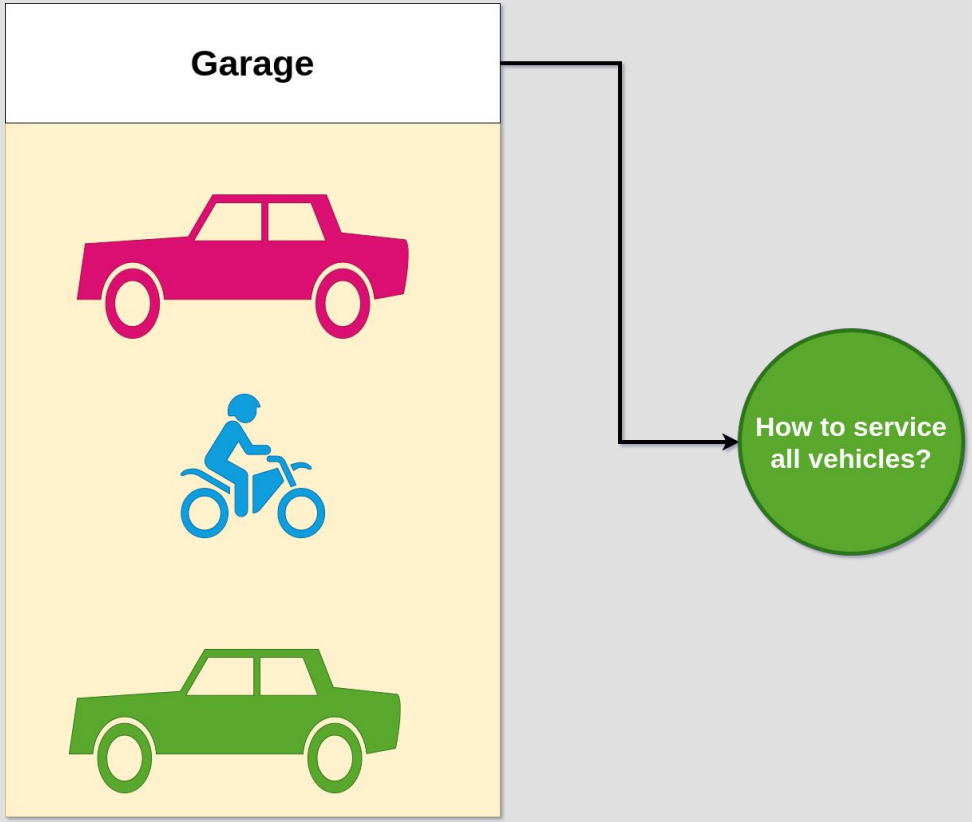
What can a bike do?

```
type bike struct {
    model    string
    engine   string
    fuel     int
    distance float32
}
```

- Drive
- Get Serviced
- Refuel

```
func (b bike) drive() {
    for b.fuel > 0 {
        b.fuel -= 1
        b.distance += 12.5
    }
}

func (b bike) service() int {
    if b.engine == "b1" {
        return 500
    } else {
        return 200
    }
}

func (b bike) refuel(fuel int) {
    b.fuel += fuel
}
```

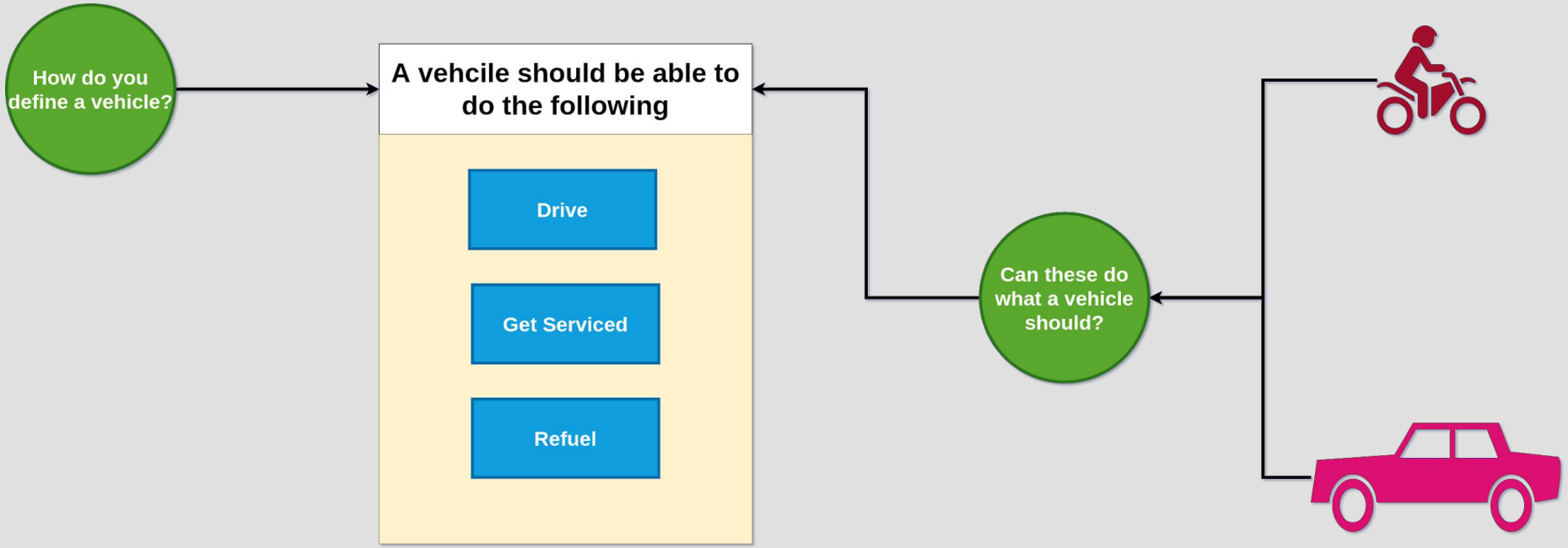
Problem



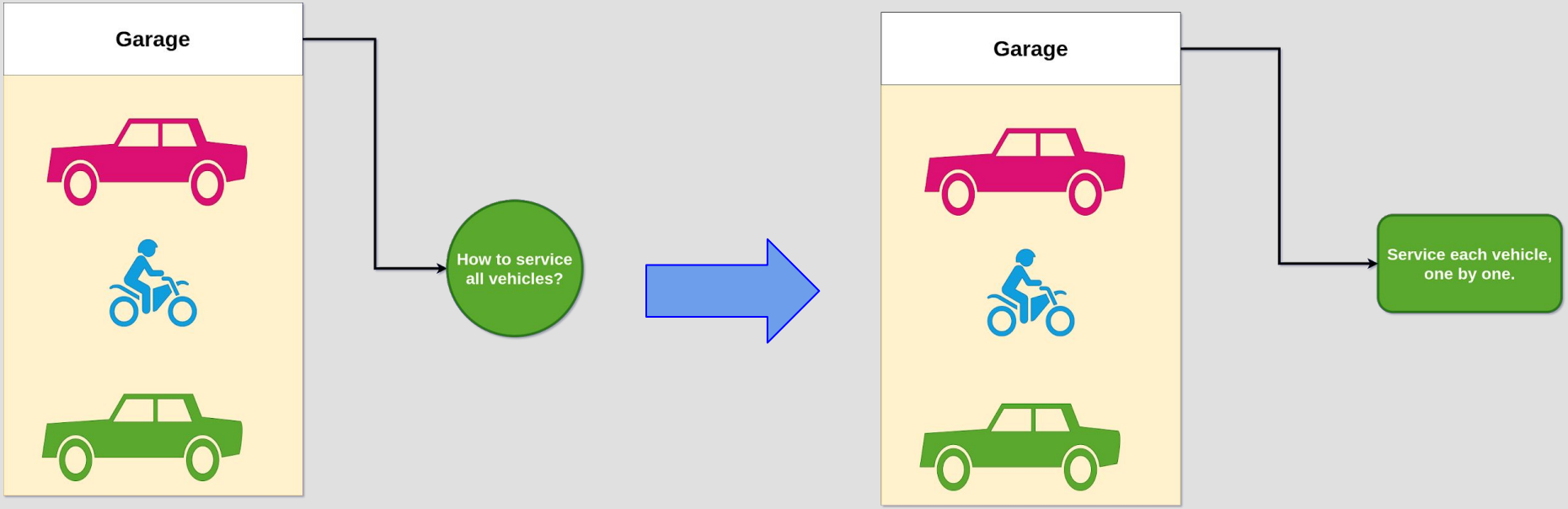
Let’s go back a bit

Can both of these be classified as a vehicle?

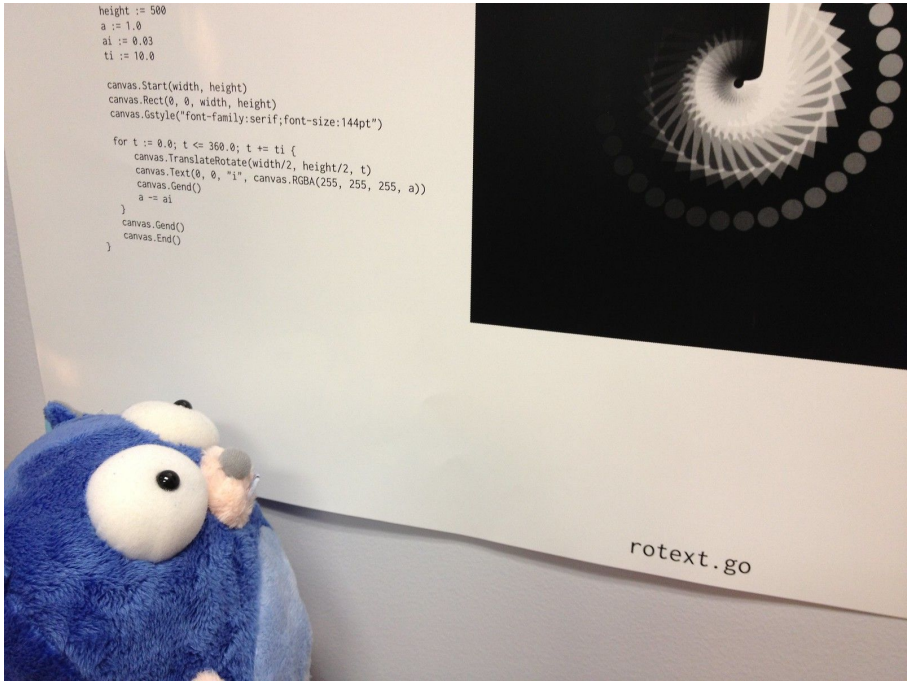
Defining “Vehicle”

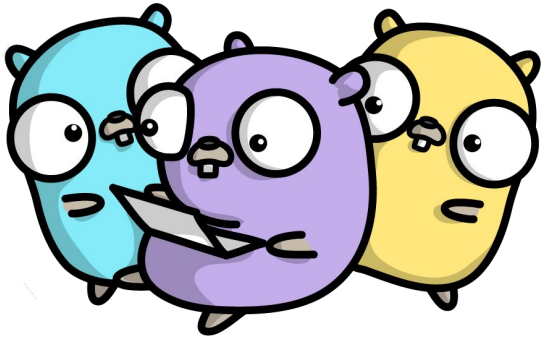


The Solution



Let’s now go take a look at code





Thank You!

Source Code and Slides available at:
github.com/Gituser143/PESU-IO-Go