— Objects First

↕

— Objects Later

Other languages allow you to define your own type.

list the atomic components that comprise or make up type

```
struct time_v1 {
    int hours;
    int minutes;
    int seconds;
}        In C
```
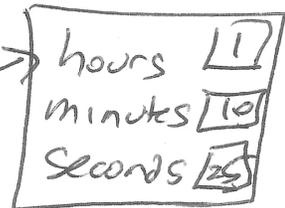

We must understand **How** the type stores data to use functions that manipulate it.

$$f(time\_v1 * ptr)\{$$

$\frac{f}{ptr}$

}

DATA ABSTRACTION

I should be able to USE a variable
of a new type w/o understanding
How it's stored.

Class is a blueprint for not
only a new type, but also the
behaviors specific to that type

Same
as
struct { 1. Instance Variables (components of object)

2. Constructor(s) — these build the object

3. Instance Methods — carry out operations on the object

Time_VI

add Time
~~so~~ difference } don't make obj

increase Time
decrease Time } do make obj

3 constructors
h, m, s

S

main
t1 →

2
30
15

add
this
~~other~~ time 2

t2 →

1
50
39

4, 20, 52

Time t3 = t1.add(t2)
− v1
t3 ⟶

4
20
52