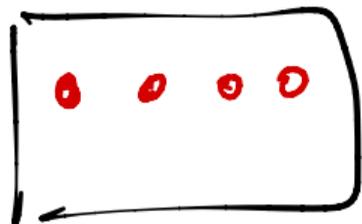


Lecture 10, 12 February 2026
Rust: structs, methods, generics, traits, lifetimes

Madhavan Mukund, S P Suresh

Programming Language Concepts
January–April 2026

ADT



class → objects

Hierarchy of classes

Rust

Not O-O

Store values

Associated functions

Public/private

Python - no public/private

- add values

Python Rust Java
|
No public/private
Type integrity

struct = "record"
tuple with named
components

Separately, associate
"methods" or functions
with structs

Referencing

`r: Rect { width
height }`
`let rect = &r;`

Dereferencing

`*rect.width` \rightarrow `rect.width`
 \uparrow
what rect points to

Java \rightarrow Type variables $\langle T \rangle$, $\langle S \rangle$ etc
 $\forall T$ $\forall S$

Rust is similar

Rust's version of a Java interface
Capability

Lifetimes are synthetic

```
{ let x = 5;
```

```
  { let y = 6;
```

```
    let a = 7;
```

```
  }
```

```
}
```

| Lifetime



Lifetime
not in scope

fn f(&p, &q) →

