

OOPS Concepts

Object Oriented Programming

CodeHelp

Why OOPS?

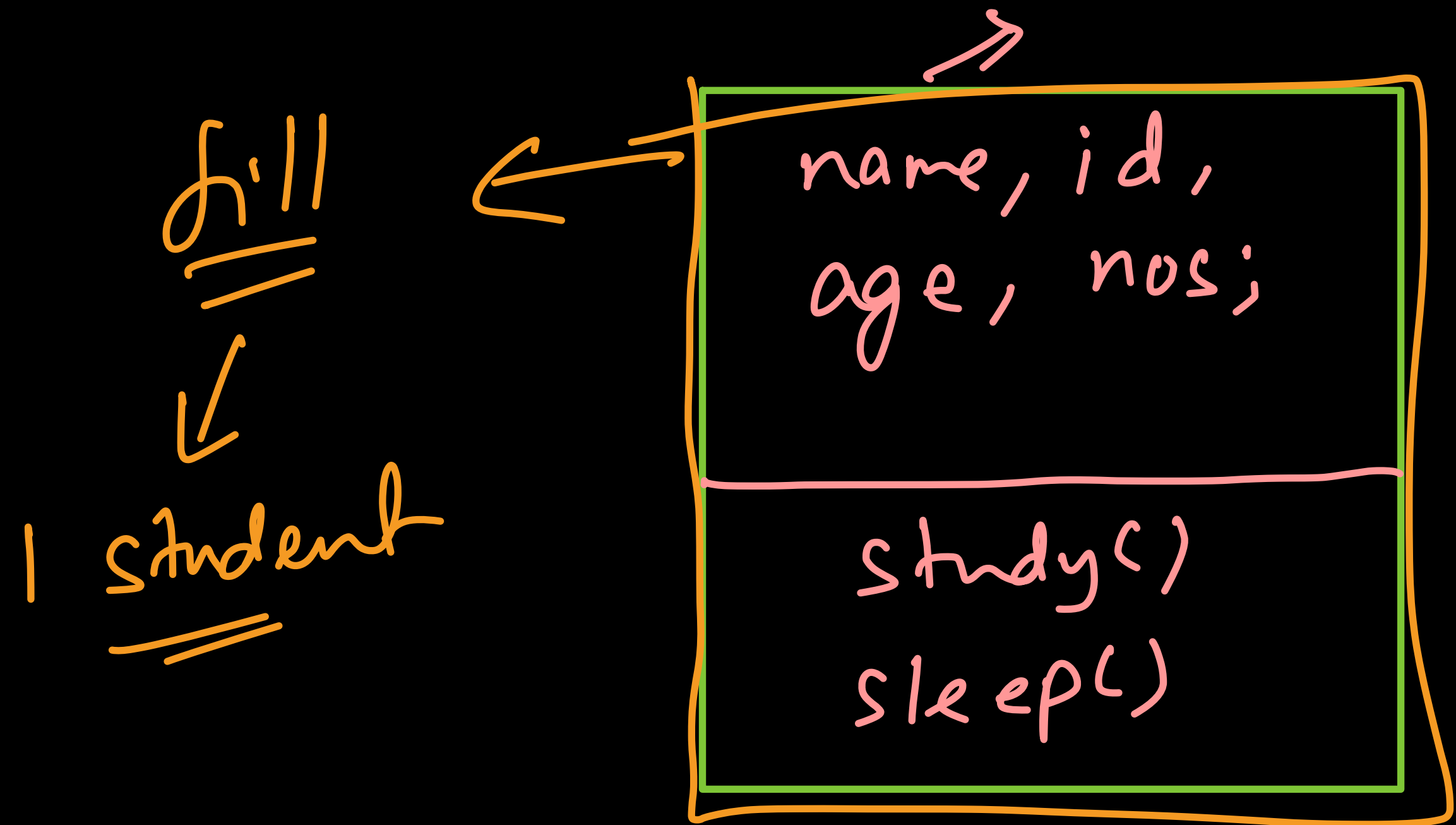
1. OOPS was developed because of limitations were discovered in earlier approaches to programming.
2. To appreciate what OOPS, Let's understand the limitations of earlier approach.



Multiple Students Example - Messy Code

1. How will you model a program having 100s of students, and each students have their own properties, behaviours and something to hide?

⇒



Multiple Students Example - Messy Code

1. How will you model a program having 100s of students, and each students have their own properties, behaviours and something to hide?

Student int main()

string School = 'DAV'

study() ✓
bunk() ✓
sleep() ✓

1000s { id, name, mos1, age1
: |
: |
: |
: |
}

⇒ Teaching()
⇒ flying()

3

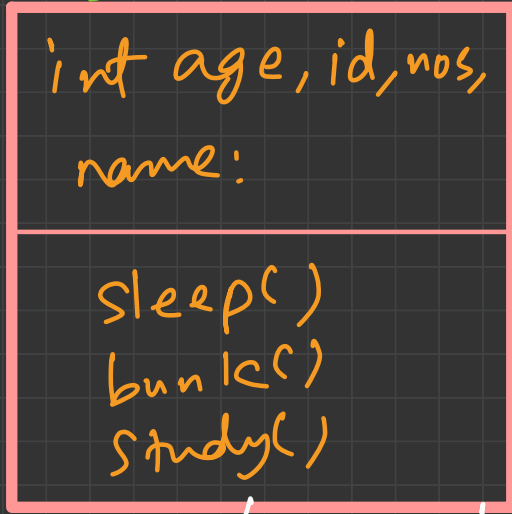
Drawbacks of Functional / Procedural Programming

1. Does not model real world problem very well.
2. If a new data is added, all the functions needs to be modified to access the Data.
3. Global data is accessible to all the functions.
4. No clear boundaries and well definition of code.
5. No Modularity: Functional programs can become monolithic and difficult to maintain as they grow in complexity.

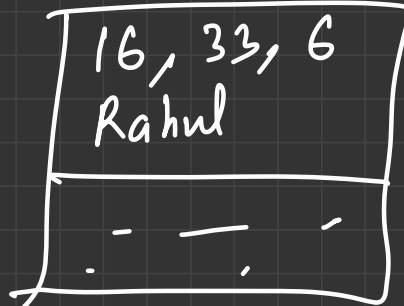
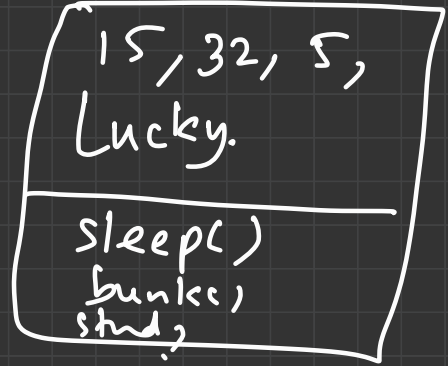
What is OOP?

1. Programming is used to solve real-world problems, how can we model real-world systems with programming languages.
2. A Programming Style, involves dividing a program into pieces of objects that can communicate with each other.
3. Objects based coding style, in which each object (aka, real-world entity) has its own attributes and behaviour.
4. Fundamental Idea is to combine into single unit, both data and behaviour, that will promote Modularity.
5. **OOP promotes modularity** by encapsulating data and behavior within objects. This modular approach enhances code reusability and maintainability, as objects can be reused in different parts of the program.
6. OOP is LIFE (We'll understand this on-the-go)

Student



S1

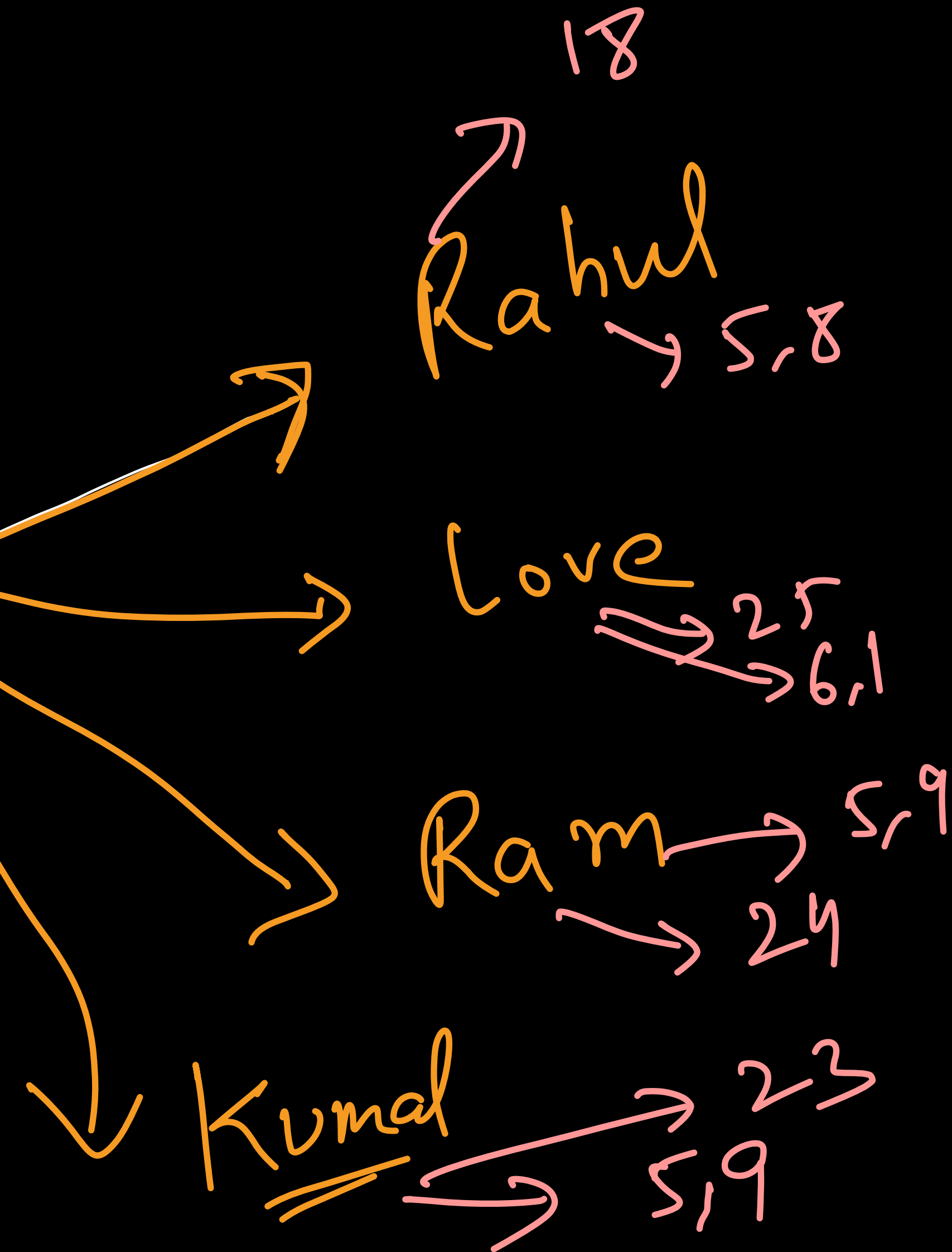
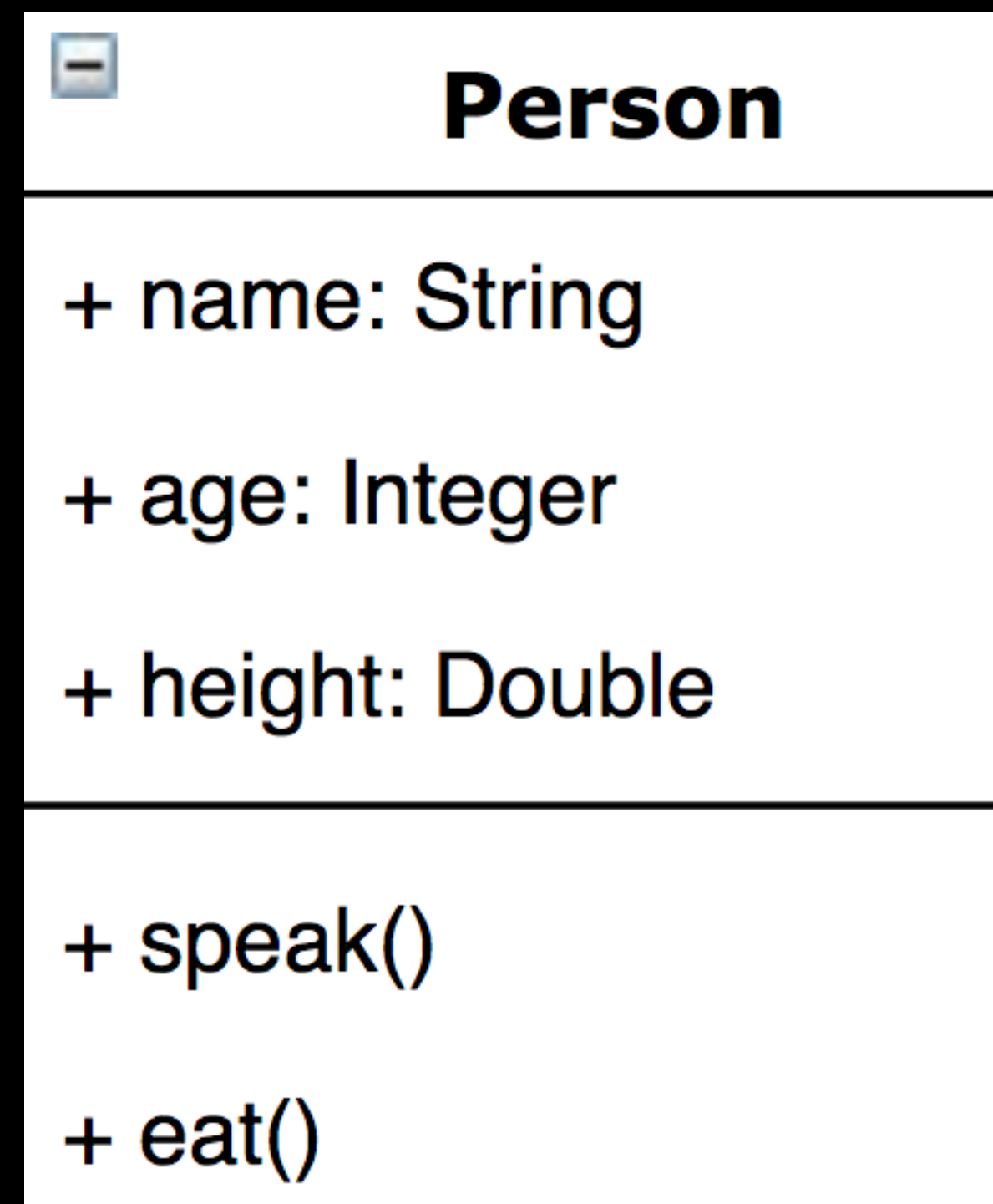


Objects and Classes.

1. Real world entities like cars, person, students, building etc., they all have some state and behaviour.
2. For e.g., a Student named Rahul, is an real world entity, in programming, he is an object.
3. What defines, how would an object look like?, there must be a Blueprint i.e., Class.
4. Hence, Object is an instance of a Class.

Attributes and Behaviour

1. Attributes are state, properties of an object.
2. Behaviour is methods / functions that an object can perform.



So...

1. Object Oriented Programming (OOP) is a programming paradigm focused on implementing real-world objects.
2. The identification of code objects similar to real-life objects and structuring code using classes and objects signifies the use of OOP principles.
3. Classes and objects serve as the fundamental building blocks of the OOP concept.
4. Major Object Oriented languages include C++, Java, and Javascript.

⇒ Integer a, b;
↓
Data Type

Class ← Student s1; ✓
↓
datatype Student s2; ✓

So...

1. Object Oriented Programming (OOP) is a programming paradigm focused on implementing real-world objects.
2. The identification of code objects similar to real-life objects and structuring code using classes and objects signifies the use of OOP principles.
3. Classes and objects serve as the fundamental building blocks of the OOP concept.
4. Major Object Oriented languages include C++, Java, and Javascript.