CODING
SHUTTLE

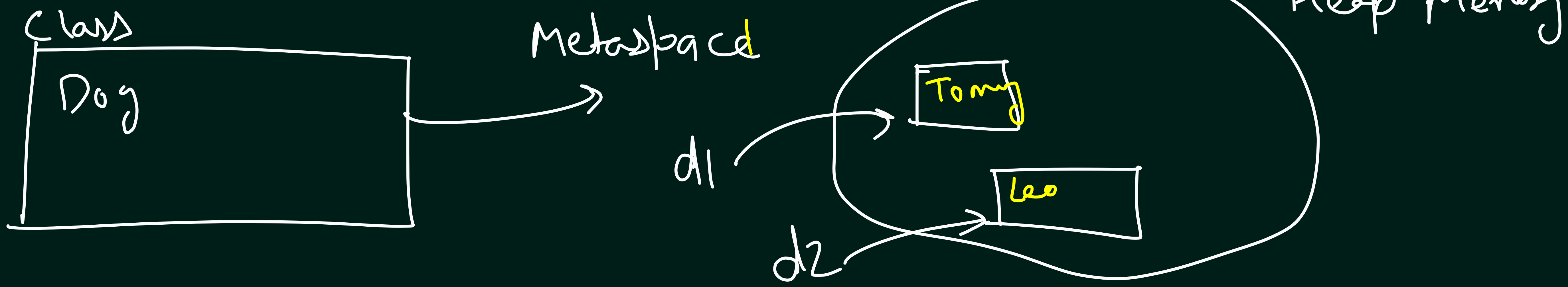
OOPS - 1

In This Lecture

1. Classes & Objects ✓
2. Constructors ✓
3. Method + Constructor Overloading ✓
4. this keyword in Java ✓

Classes & Objects

1. Class is a blueprint which defines some properties and behaviors. An object is an instance of a class which has those properties and behaviours attached.
2. A class is not allocated memory when it is defined. An object is allocated memory when it is created.
3. Class is a logical entity whereas objects are physical entities.
4. A class is declared only once. On the other hand, we can create multiple objects of a class.

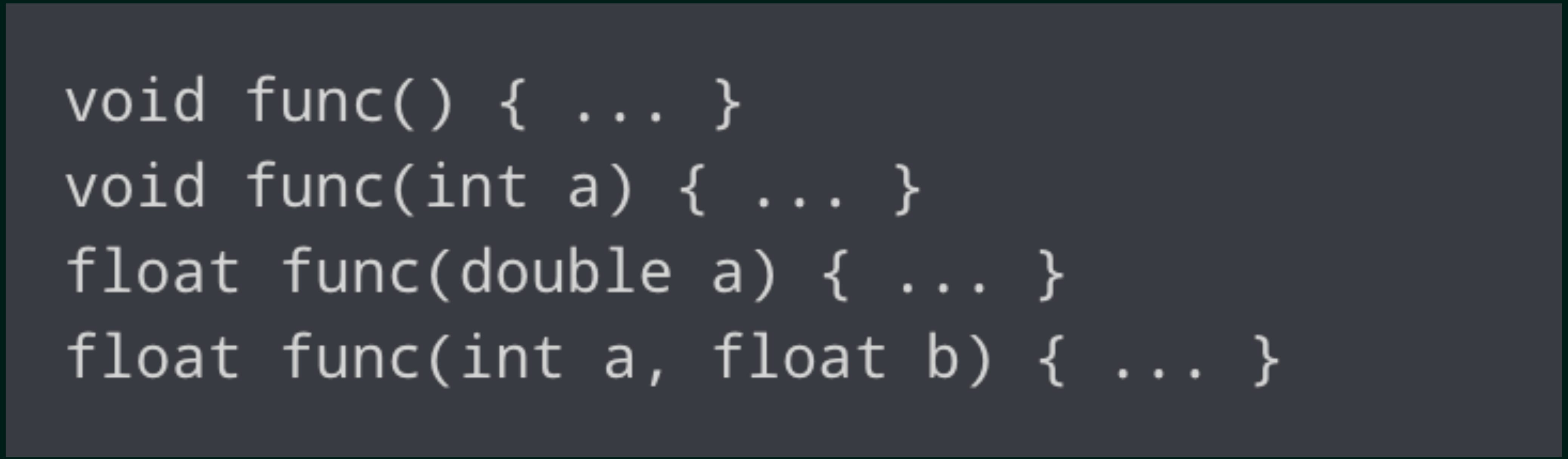


Classes & Objects

5. A class is a way to arrange data and behavior information. It is a template that must be implemeted by its objects.
6. A class can also be seen as a user-defined data type where any object of defined data type has some predefined properties and behaviors.

Method Overloading

1. Two or more methods can have the same name inside the same class if they accept different arguments. This feature is known as method overloading.
2. Method overloading is achieved by either:
    a. changing the number of arguments.
    b. or changing the data type of arguments.
3. It is not method overloading if we only change the return type of methods. There must be differences in the number of parameters.



```
void func() { ... }
void func(int a) { ... }
float func(double a) { ... }
float func(int a, float b) { ... }
```

Constructors

1. Constructors are invoked implicitly when you instantiate objects.
2. The two rules for creating a constructor are:
    a. The name of the constructor should be the same as the class.
    b. A Java constructor must not have a return type.
3. If a class doesn't have a constructor, the Java compiler automatically creates a default constructor during run-time. The default constructor initializes instance variables with default values.
4. Default Constructor - a constructor that is automatically created by the Java compiler if it is not explicitly defined.
5. A constructor cannot be abstract or static or final.
6. A constructor can be overloaded but can not be overridden.

The this keyword

In Java, this keyword is used to refer to the current object inside a method or a constructor.

We mostly use this keyword to remove any Ambiguity in Variable Names. We can also use this to invoke methods of the current class or to invoke a constructor of the current class.