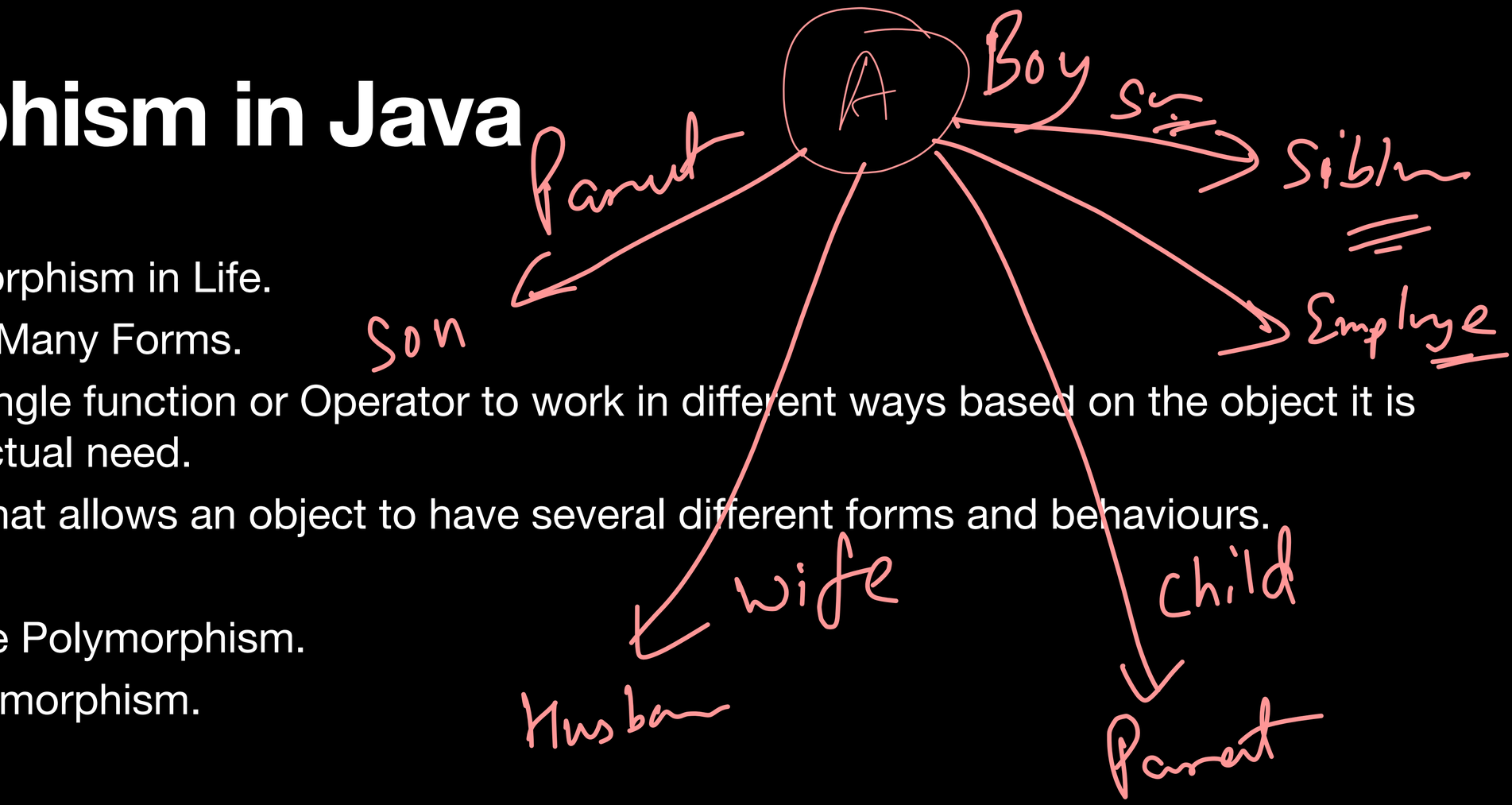
Polymorphism in Java



1. Similar to Polymorphism in Life.
2. Polymorphism = Many Forms.
3. The ability of a single function or Operator to work in different ways based on the object it is acting upon or actual need.
4. A phenomenon that allows an object to have several different forms and behaviours.
5. Types
    1. Compile Time Polymorphism.
    2. Runtime Polymorphism.

Static Polymorphism

1. Aka, Compile Time Polymorphism.
2. Types
    1. Method Overloading
    2. Operator Overloading

Method Overloading

1. Overloading occurs when a class contains multiple methods sharing a name but differing in argument count or argument type.

```
class Calculator {
    // Method to add two integers
    int add(int a, int b) {
        return a + b;
    }
    // Overloaded method to add three integers
    int add(int a, int b, int c) {
        return a + b + c;
    }
}
```

Method Overloading

Operator Overloading

1. Java does not support user-defined operator overloading but internally overloads certain operators (e.g., + for String concatenation and integer addition).

Runtime Polymorphism

1. **Function Overriding** - A Subclass can provide a specific implementation of a method that is already defined in its superclass.
2. **Dynamic Method Dispatch** [Upcasting] - A superclass reference variable can refer to a subclass object and dynamic dispatch is used to resolve the called method at runtime.

downcasting ≡ Child c = Parent

Runtime Polymorphism