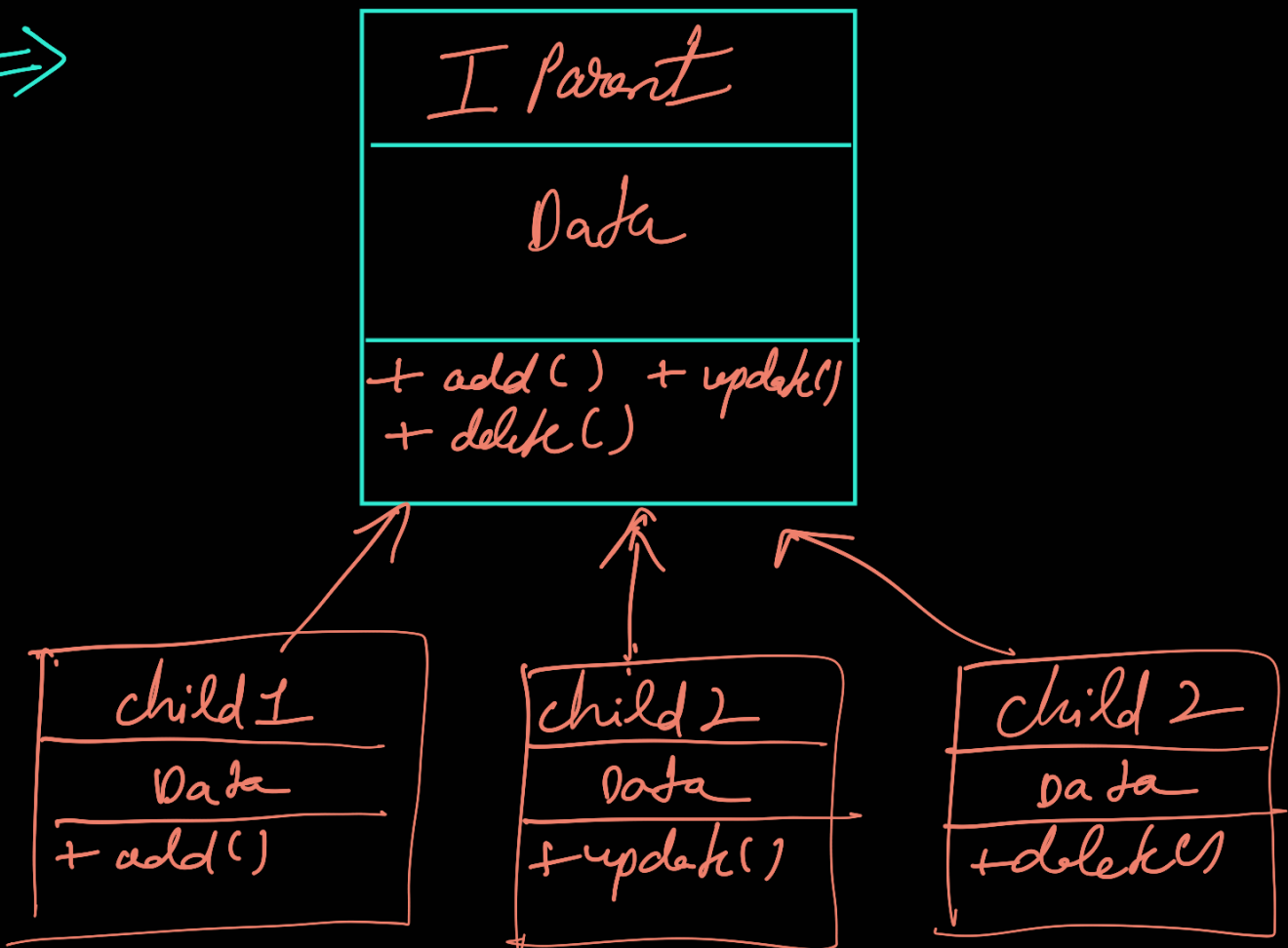# 4. Interface Seggregation Principle

⟹ Prevent classes from relying on things that they don't need.

⟹ Don't bloat your class with many functions.

⟹ This is kind of extension of "Single Responsibility Principle"

⟹

```
┌─────────────────────────────┐
│         I Parent            │
├─────────────────────────────┤
│          Data               │
├─────────────────────────────┤
│ + add ( )   + update()      │
│ + delete ( )                │
└─────────────────────────────┘
```

```
┌──────────────┐   ┌──────────────┐   ┌──────────────┐
│   child 1    │   │   child 2    │   │   child 2    │
├──────────────┤   ├──────────────┤   ├──────────────┤
│    Data      │   │    Data      │   │    Data      │
├──────────────┤   ├──────────────┤   ├──────────────┤
│  + add()     │   │ + update()   │   │ + delete()   │
└──────────────┘   └──────────────┘   └──────────────┘
```

↓
This only wants
add(), but has
to implement other
two methods as empty

& So, there is need to seggregate
interfaces