1 OOP concepts:

* Encapsulation (Entity classes with private variables & getters setters)
* Abstraction (methods in Interfaces)
* Inheritance
* Polymorphism (runtime overriding & compile time overloading)

In overriding return type of overridden method can be subtype of return type of parent method (parent can reference child obj but vice versa not true), else has to be with same return type.

* Example of runtime poly : In controller we inject interface (service) not its implementation class, & we can have 2 Impl of this interface & their methods could be called.
* Parent p = new Child(), p.m1() will call child m1 method unless m1 is not there/overridden in child.
* We cannot override static (@override annotation will show error, if we try to override m1 without annotation & call, still parents method will be called & method without annotation will get hidden) & private (they not visible to child class) methods.