② Runtime / dynamic polymorphism

```
class Parent {
    int a;
}

class Child extends Parent {
    int b;
}
```

```
Parent p = new Parent();
```

reference variable → new object

```
Child c = new Child();
Parent obj = new Child();
```

✓ `obj.a = 10;`

✗ `obj.b = 10;`

Function Overriding

Static Polymorphism	Dynamic Polymorphism
Overloading	Overriding
Early binding	Late binding
compile time	run time

final keyword

↳ ① Variables

final int a; → can only assign value once

→ ② Functions

↳ cannot be overriden

→ ③ Classes

↳ cannot extend

---

③ Encapsulation

↳ binding all the data members and member functions in class together

→ Wrapping everything in class

④ Abstraction

→ Hiding all the unnecessary details and show only the required part.

Access. Modifiers

Abstract class

Interface

Access Modifiers / Specifiers

↳ specifies access / permission

1. Public
2. Protected
3. Default
4. Private

	Class	Package	Sub class in diff package	Anywhere / diff package
Public	✓	✓	✓	✓
Protected	✓	✓	✓	
Default	✓	✓		
Private	✓	×	×	×