



ABSTRACTIONS

OOP - Java

Ken + Willmen





Apa Itu Abstraction?

- Menyembunyikan detail kompleks, menampilkan fitur penting saja
- Menyederhanakan interaksi dengan objek
- Fokus pada "apa" yang dilakukan objek, bukan "bagaimana" caranya
-



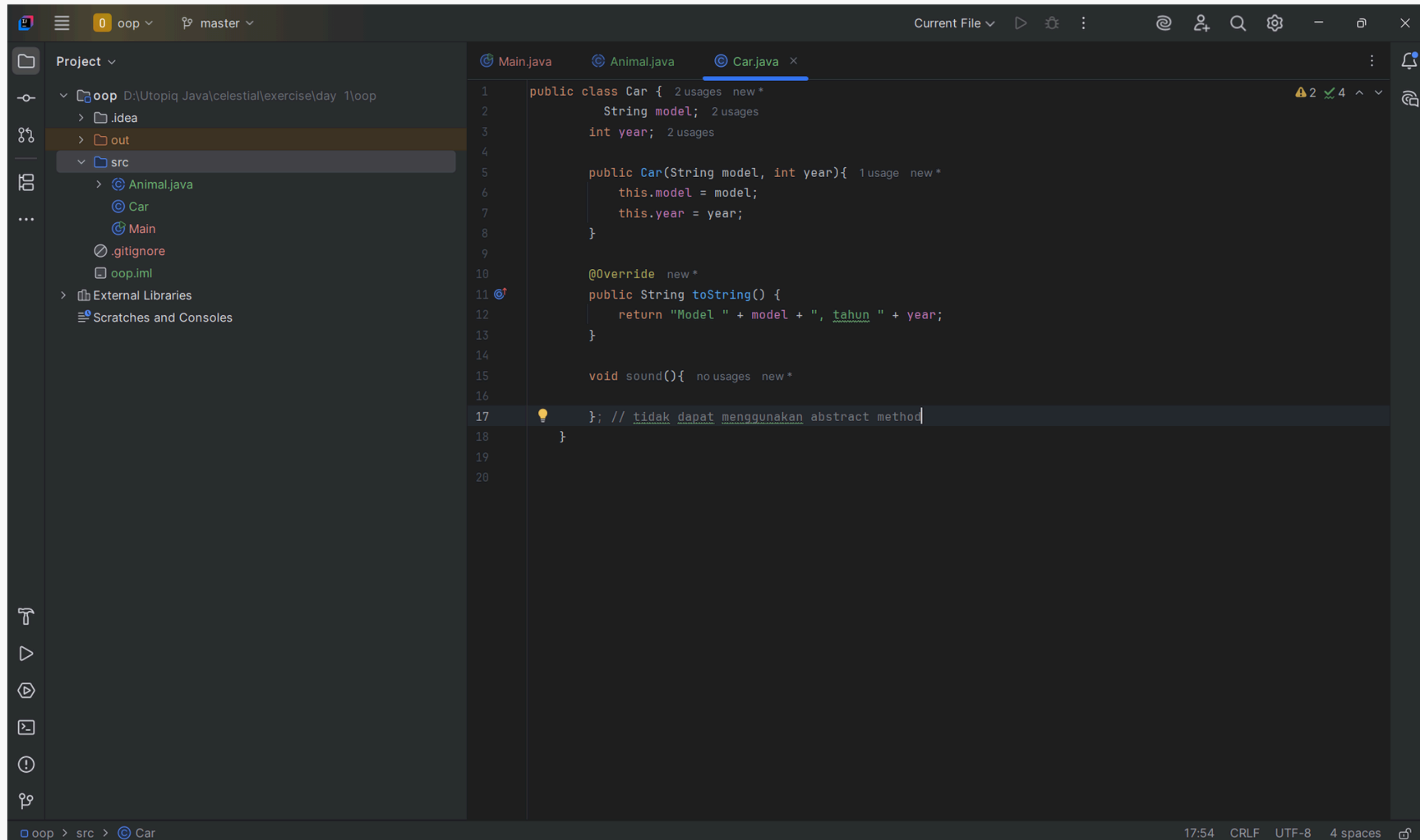
Untuk Apa?

- Mewajibkan subclass mengikuti struktur yang sama
- Mendukung polimorfisme (satu antarmuka, banyak implementasi)
- Mengurangi duplikasi kode & mempermudah pemeliharaan
- Biasanya sebagai template

Tabel Perbandingan

Aspek	Class Biasa	Abstract Class
Dapat diinstansiasi?	✔ Ya, bisa langsung dibuat objek	✗ Tidak bisa dibuat objek langsung
Memiliki method abstrak?	✗ Tidak boleh	✔ Boleh punya method abstrak (tanpa isi)
Tujuan	Digunakan untuk membuat objek nyata	Digunakan sebagai kerangka (template)
Method konkret?	✔ Semua method harus ada isi	✔ Boleh punya method dengan dan tanpa isi
Keyword	class	abstract class
Contoh penggunaan	class Car {}	abstract class Animal { abstract void sound(); }

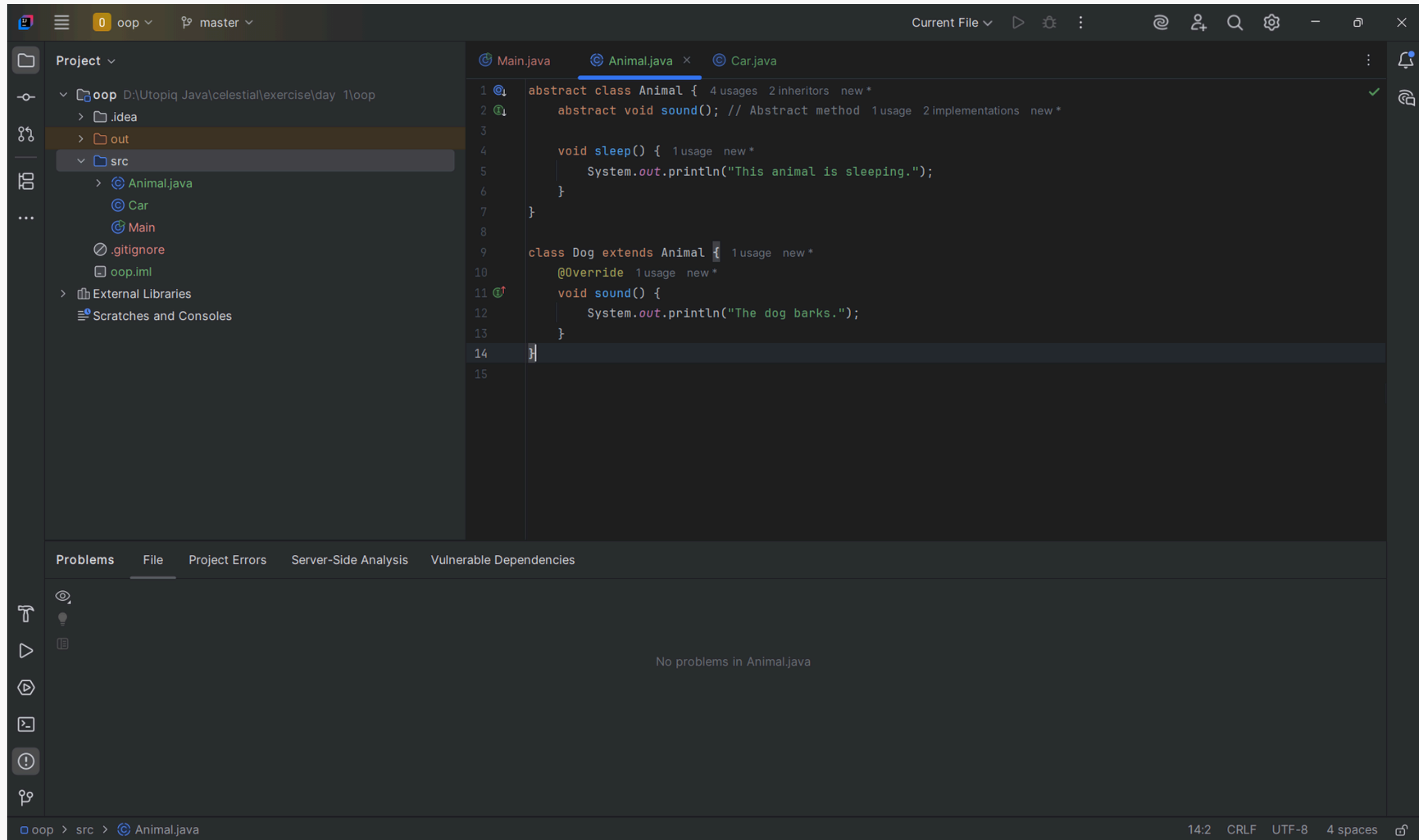
Class Biasa



```
1 public class Car { 2 usages new *
2     String model; 2 usages
3     int year; 2 usages
4
5     public Car(String model, int year){ 1 usage new *
6         this.model = model;
7         this.year = year;
8     }
9
10    @Override new *
11    public String toString() {
12        return "Model " + model + ", tahun " + year;
13    }
14
15    void sound(){ no usages new *
16
17    }; // tidak dapat menggunakan abstract method
18 }
19
20
```

oop > src > Car 17:54 CRLF UTF-8 4 spaces

Abstract Class



Main

