



# Жизненный цикл компонента





# LifeCycle

<https://angular.io/guide/lifecycle-hooks>

Набор методов, которые вызываются на разных этапах работы компонента (от создания до уничтожения)

Нужно знать, для правильной инициализации переменных (чтобы не было ошибки **undefined**)

Без полного знаний как работает Angular - нет смысла запоминать все методы. Только по мере их использования (понимания)

constructor

ngOnChanges

ngOnInit

ngDoCheck

ngAfterContentInit

ngAfterContentChecked

ngAfterViewInit

ngAfterViewChecked

ngOnDestroy

```

export class TasksComponent implements OnInit, AfterViewInit {

    // поля для таблицы (те, что отображают данные из задачи - должны совпадать)
    private displayedColumns: string[] = ['color', 'id', 'title', 'date', 'priority'];
    private dataSource: MatTableDataSource<Task>; // контейнер - источник данных

    // ссылки на компоненты таблицы
    @ViewChild(MatPaginator, {static: false}) private paginator: MatPaginator;
    @ViewChild(MatSort, {static: false}) private sort: MatSort;

    tasks: Task[];

    constructor(private dataHandler: DataHandlerService) {
    }

    ngOnInit() {
        this.dataHandler.tasksSubject.subscribe(next: tasks => this.tasks = tasks);

        // датасорс обязательно нужно создавать для таблицы, в него присваиваем данные
        this.dataSource = new MatTableDataSource();

        this.refreshTable();
    }

    // в этом методе уже все проинициализировано, поэтому можно присваивать объекты
    ngAfterViewInit(): void {
        this.addTableObjects();
    }
}

```

Сначала нужно подключить соотв. Интерфейс (после этого метод можно создать автоматически, а не прописывать вручную)

Методы могут располагаться в любом месте класса (необязательно друг за другом)



# Домашнее задание

- Прочитать <https://angular.io/guide/lifecycle-hooks>