



Сайт: <https://dotnetmore.ru>

VK: <https://vk.com/dotnetmore>

Rss: <https://dotnetmore.ru/feed/podcast/>

SoundCloud:

<https://soundcloud.com/dotnetmore>

Twitter: <https://twitter.com/dotnetmore>

Telegram channel: <https://t.me/dotnetmore>

Telegram chat: https://t.me/dotnetmore_chat

DOT NET & MORE ПОДКАСТ

Test Last Test First TDD

когда применять тот или иной подход

Кугушев Александр

EPAM Systems

Lead Software Engineer

kugushew@gmail.com

www.linkedin.com/in/kugushev/

github.com/AleksandrKugushev

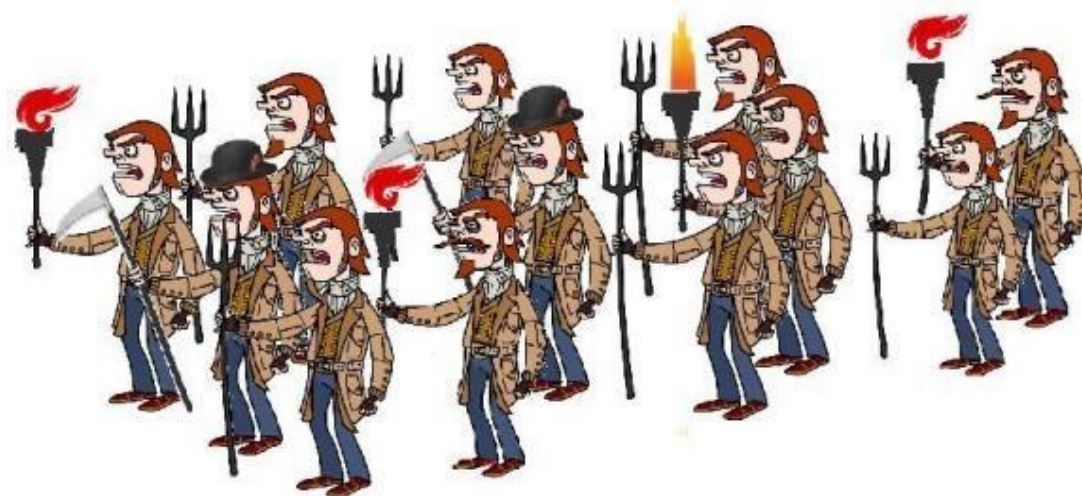




Что вы предпочитаете?



Отвертка?



Молоток?



Все инструменты необходимы

A cartoon illustration of a mob of men with exaggerated features, wearing brown coats and blue trousers. They are holding torches and pitchforks, suggesting a scene of violence or protest. The background is dark and textured.

Война

“TDD as you mean it is a great academic exercise for your free time or weekends”

- Rob Ashton

“TDD, в том виде, как вы его представляете, это отличное академическое упражнение для вашего свободного времени или выходных”

- Роб Эштон

A cartoon illustration of a mob of men in brown trench coats and hats, holding pitchforks and torches, marching towards the left. The background is dark and textured.

Война

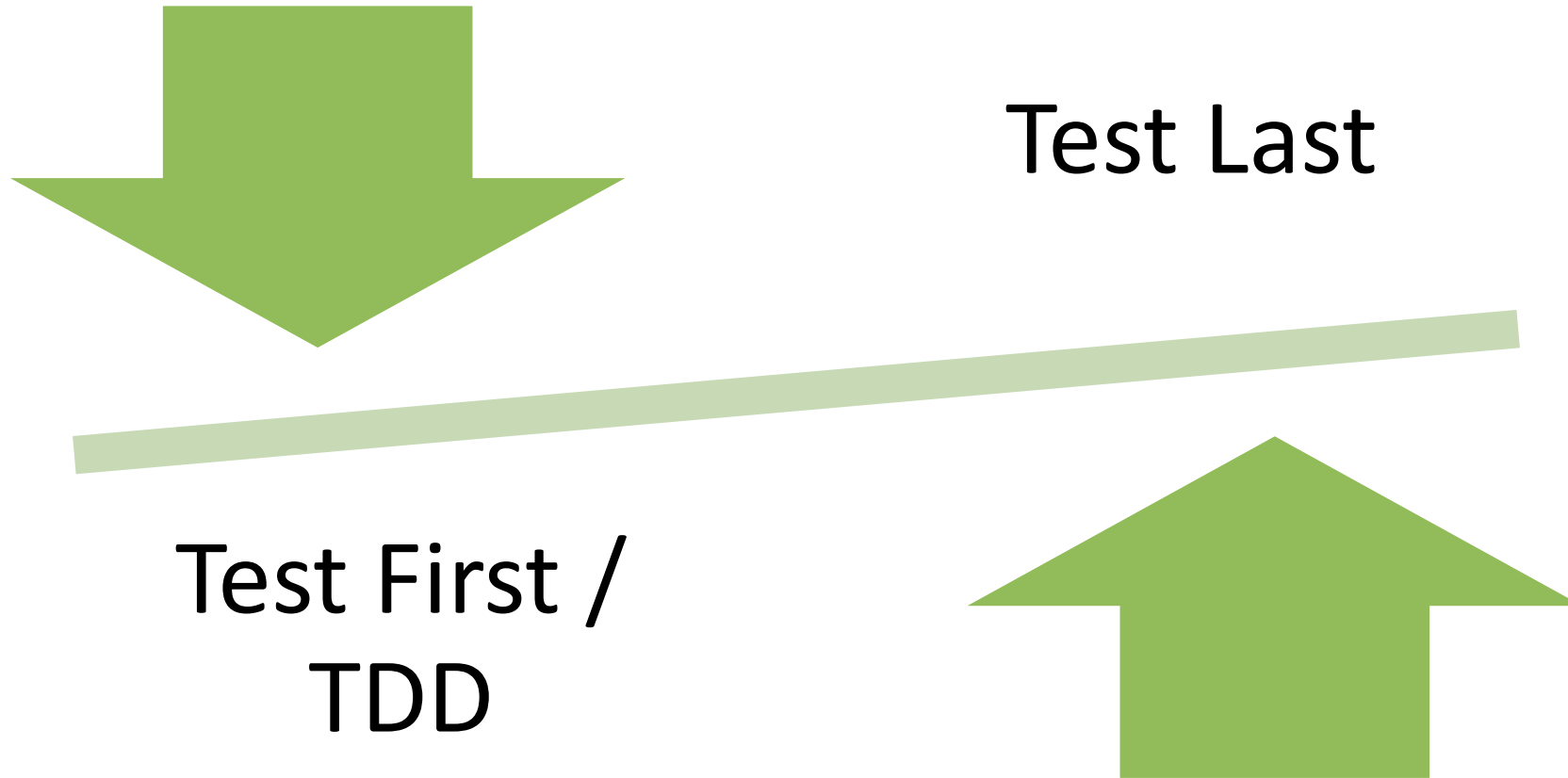
“Today, at this moment in our history, TDD is not the prerequisite of professionalism that I believe it will become.”

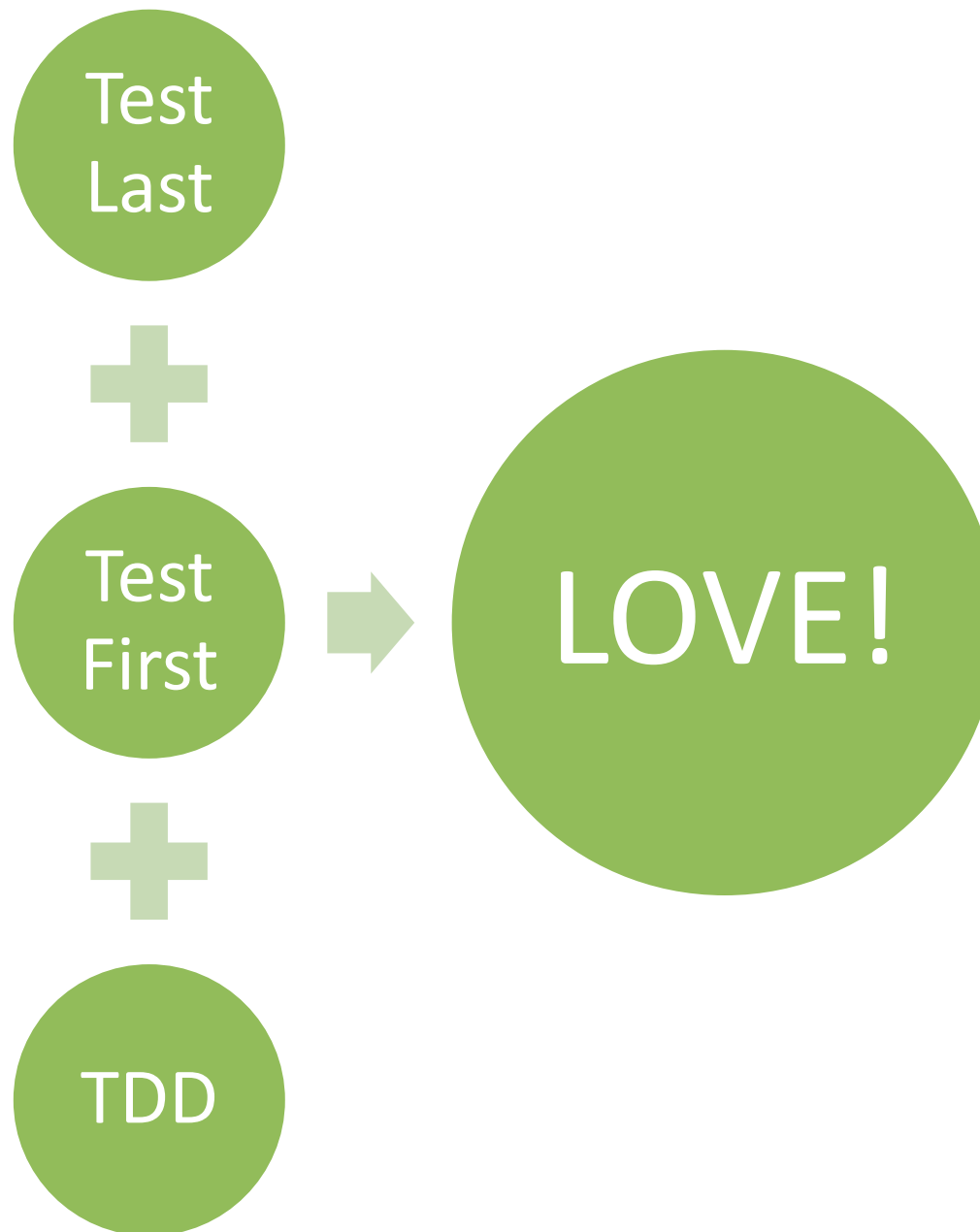
- Robert Martin

“На текущий момент нашей истории, TDD не является обязательным атрибутом профессионализма, но, я считаю, он станет таковым”

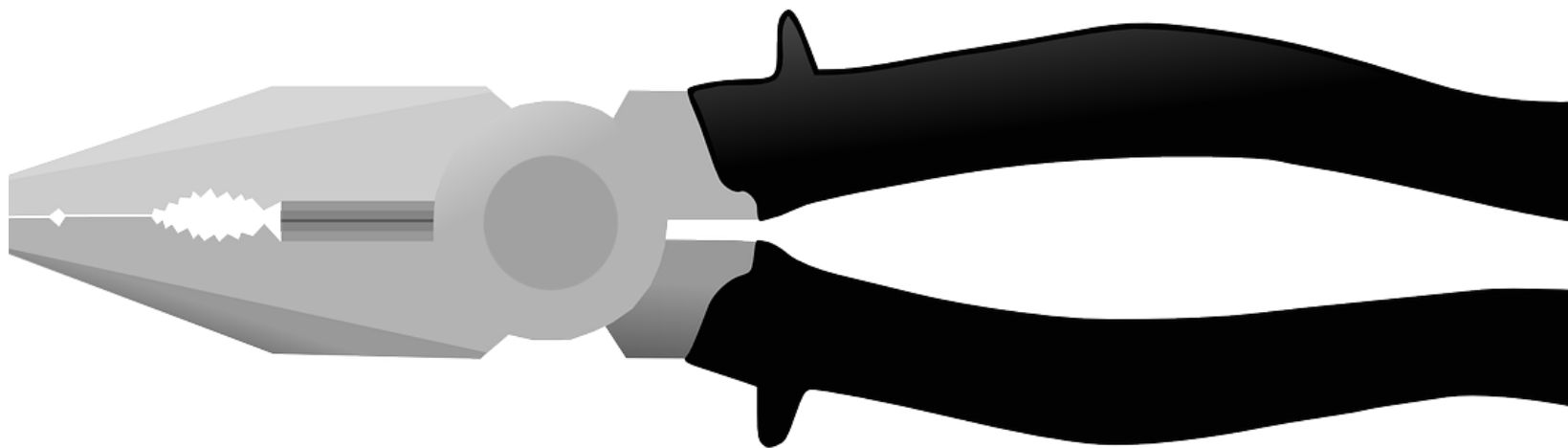
- Роберт Мартин

Test Last vs TDD





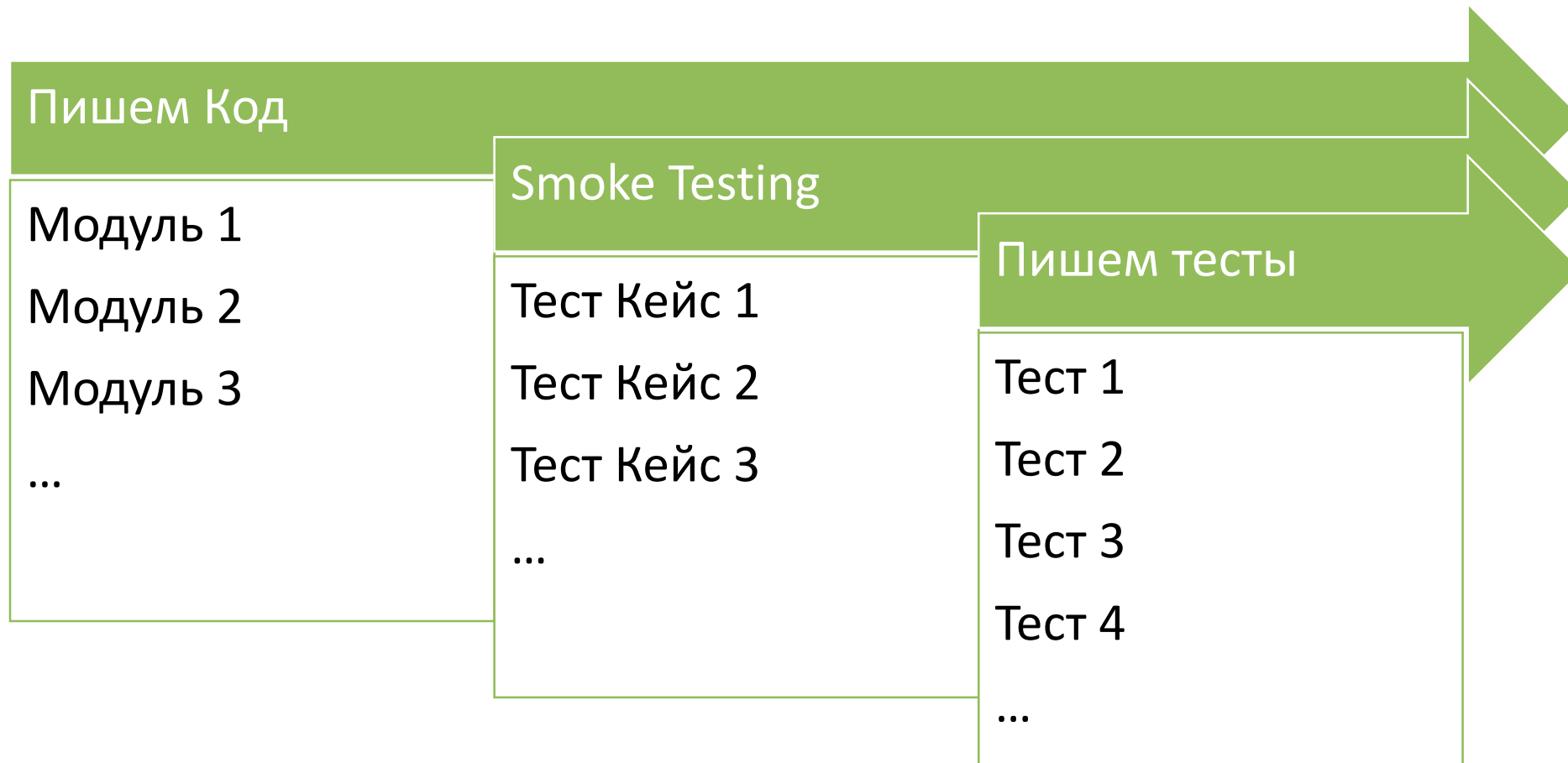
Что это?



Определимся с терминами

Часть 1

Test Last



Test Last: TODO way

```
// TODO: Test it
public void Feature1()
{
    ...
}
...
...
...
```

...

...

...

```
// TODO: Test it
public void Feature2()
{
    ...
}
```

Test Last: Jira/TFS/etc.

Feature #1

- 1 Story Point

Feature #2

- 2 Story Points

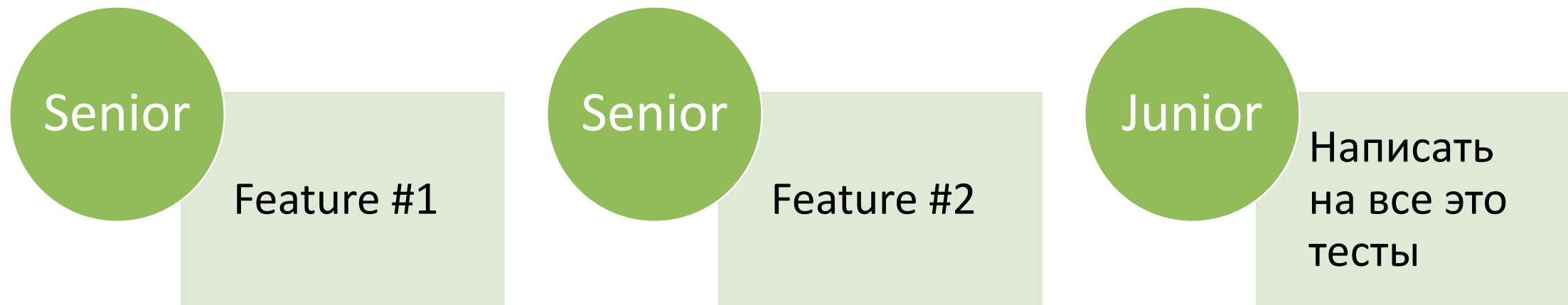
Написать на все это тесты

- 2 Story Points

Test Last: Проблемы

- Не всегда есть время на тесты

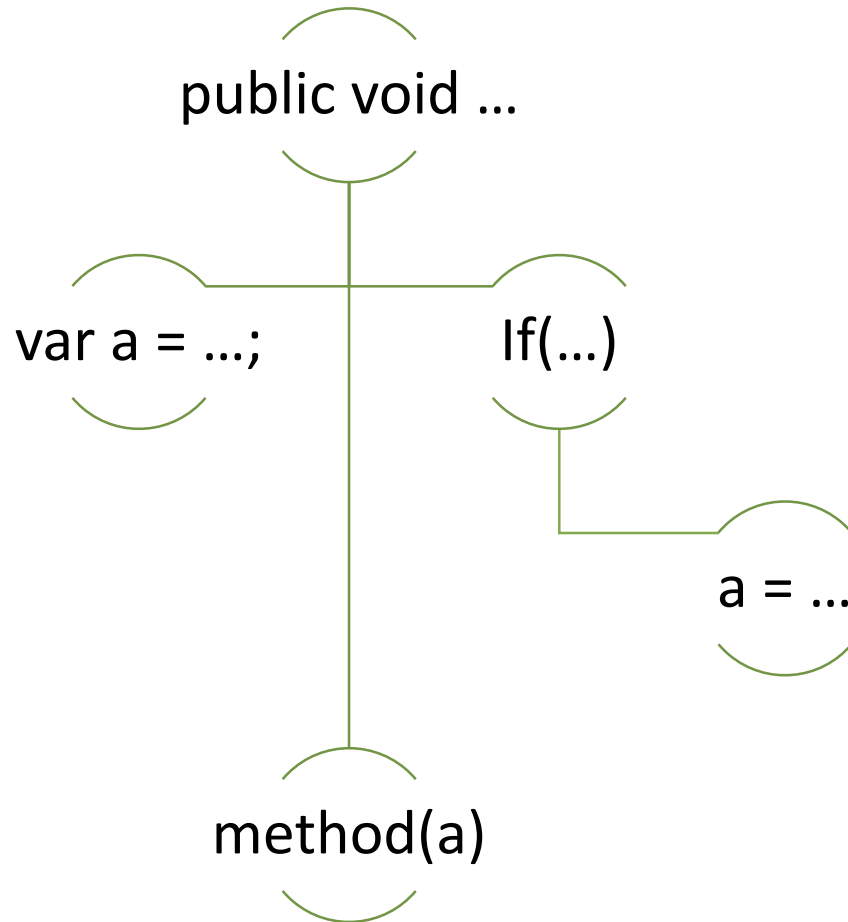
Test Last: Распределяем задачи



Test Last: Проблемы

- Не всегда есть время на тесты
- Тесты никто писать не любит

Test Last: Как мы пишем код?



Как это протестировать?

- Цикломатическая сложность
- Code Coverage
 - Можно пропустить важный case
- Test case
- Per feature
 - Мешает понимание реализации
- Слушать интуицию
 - Почему нужны manual QA?

Test Last: Проблемы

- Не всегда есть время на тесты
- Тесты никто писать не любит
- Понимание реализации влияет на тесты
 - Сложно написать достаточно тестов
 - Сложно протестировать то что нужно



Test Last: Если код уже
написан?

```
[Test]
public void Test()
{
    // arrange
    ...
    // act
    Execute.....
    // assert
    Assert.....
}
```



Test Last: Если код уже написан?

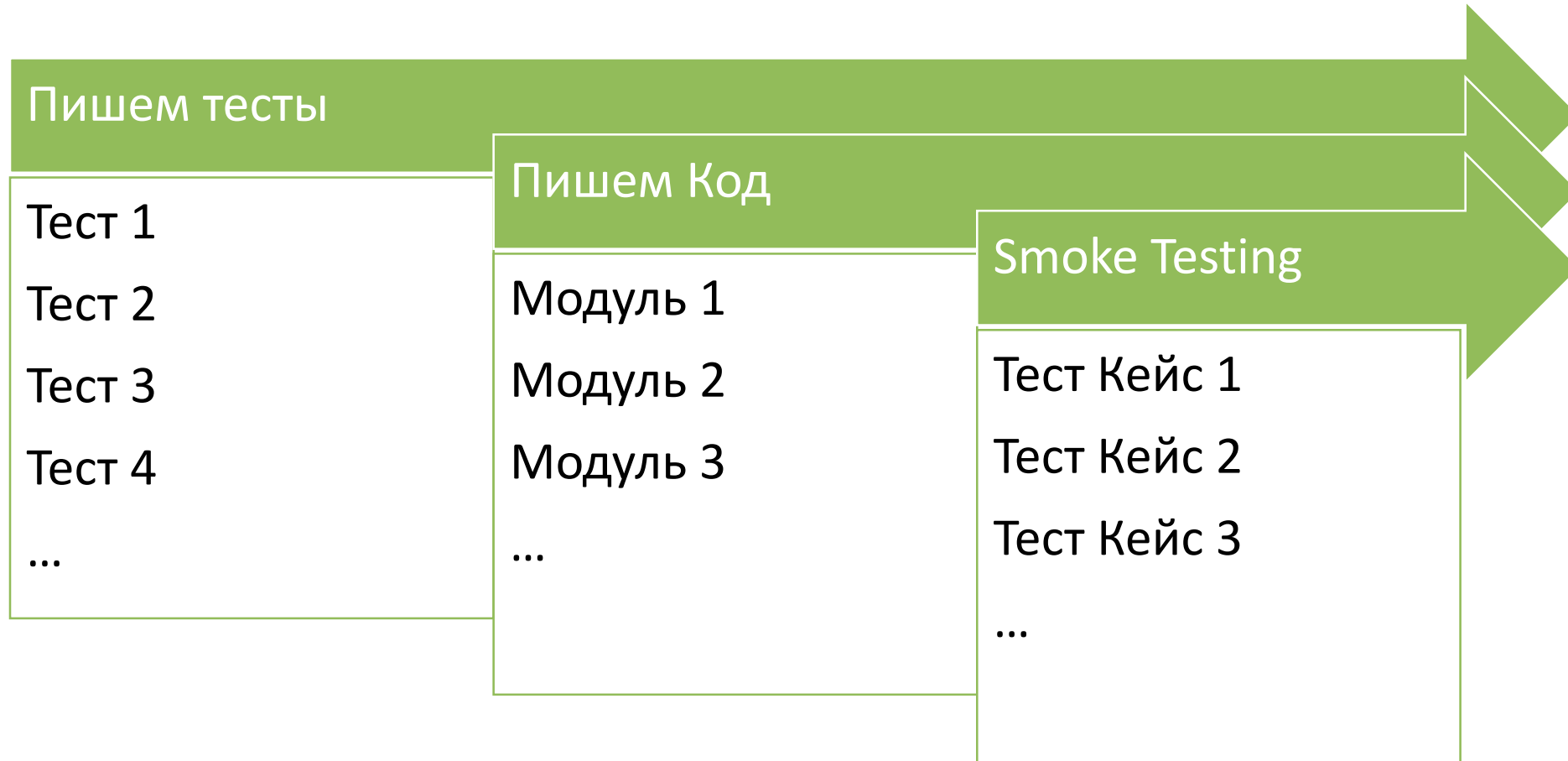
```
[Test]
public void Test()
{
    // arrange
    var a = new Stub();
    var b = new Mock();
    Setup(x => x.DoSomething(
        Is.Any<Dict.....
    Prepare(.....
    Fake.Fake.Fake(.....
    .....
    .....
    .....
    // act
    Execute.....
    // assert
    Assert.....
}
```


Test Last

- Не всегда есть время на тесты
- Тесты никто писать не любит
- Понимание реализации влияет на тесты
 - Сложно написать достаточно тестов
 - Сложно протестировать то, что сложно
- Иногда получаюся уродливые тесты



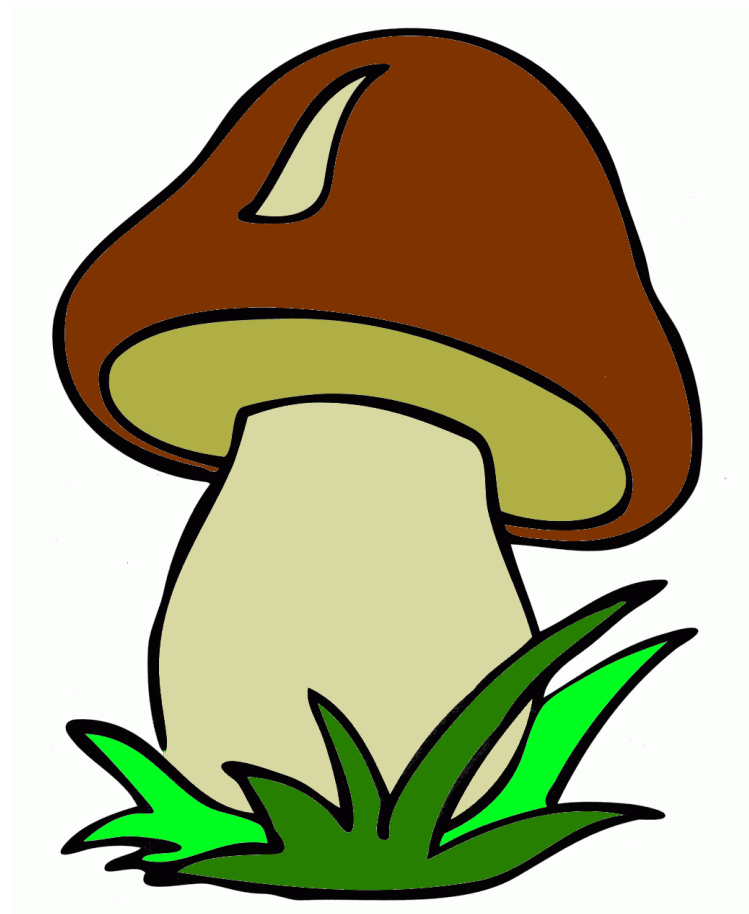
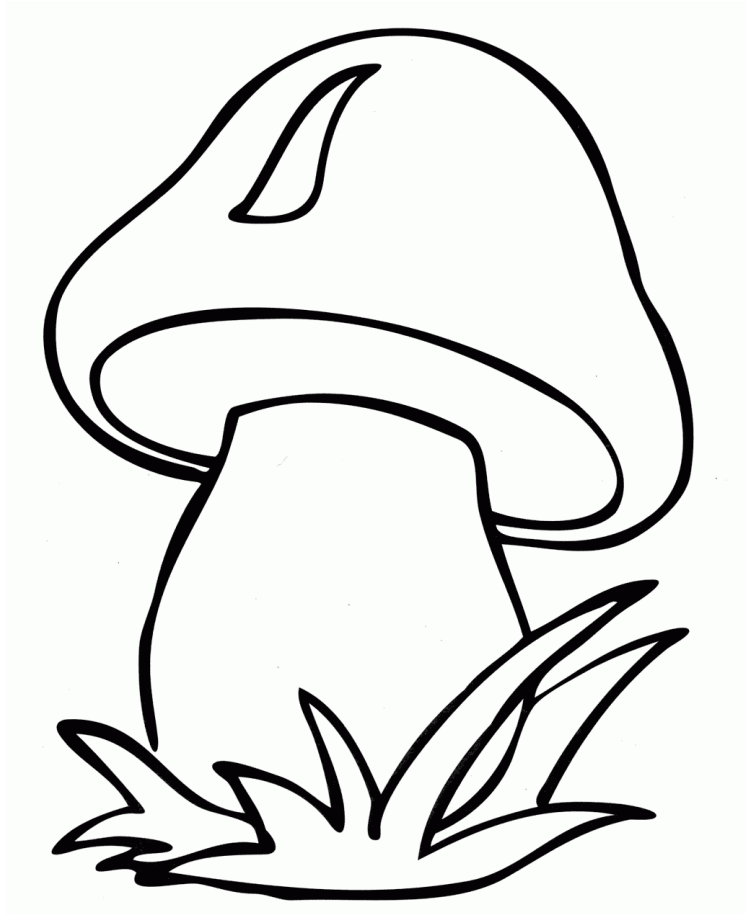
Может писать тогда **сначала** тесты?





Test First это не
TDD!

Test First



Test Last vs Test First

Test Last

- Не всегда есть время на тесты
- Тесты никто писать не любит
- Понимание реализации влияет на тесты
- Иногда получаются уродливые тесты

Test First

- На тесты есть время **всегда**
- Тесты, по-прежнему, никто писать **не любит**
- Код **не влияет** на тесты
- Тесты, **как правило**, получаются красивенькие

Test First: Проблемы

- Возможно только если есть четкая спецификация
- Иначе – пустая трата времени и сил

Test First: Цитата

“Do you know exactly what the public API of the class should be and just write the tests before the implementation? That's test-first development.”

- Jon Skeet

**Test First
Specification
first**



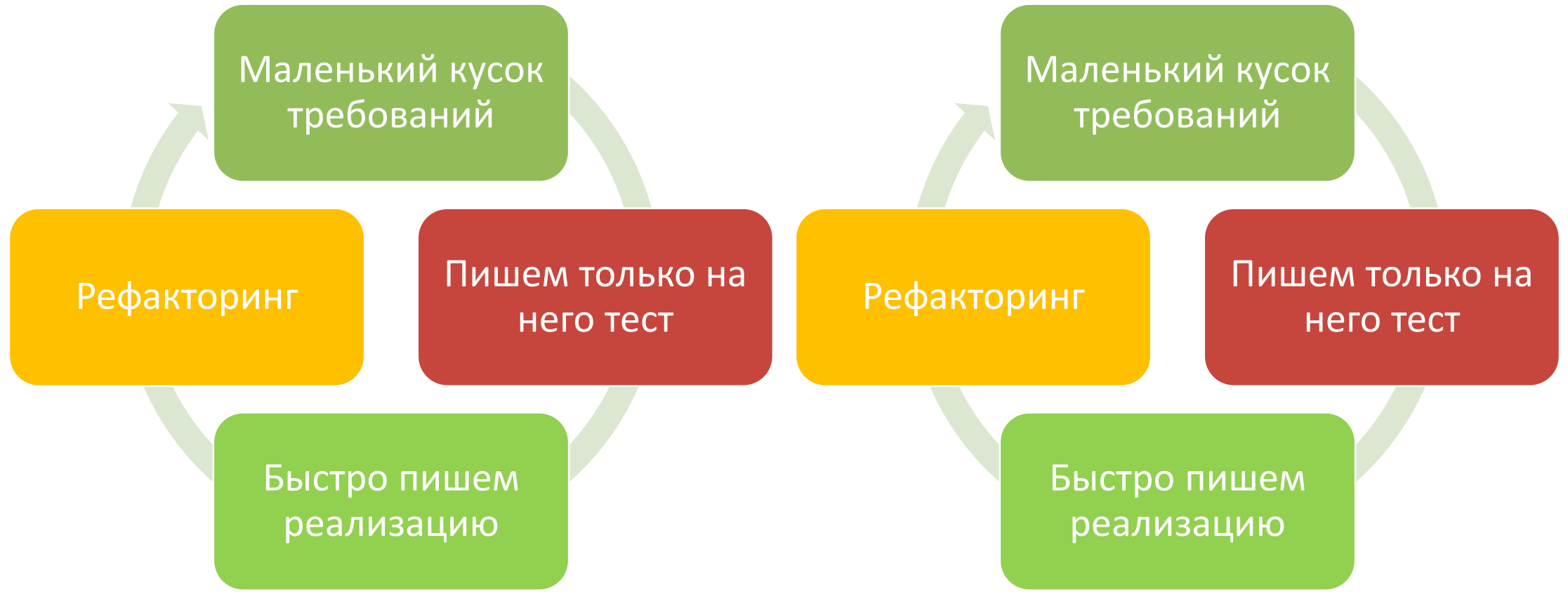
TDD: Цитата

“Do you have a vague idea of what the class (or system - this can happen at different scales, of course) should look like, then think up tests which give it the actual shape? That's TDD.”

- Jon Skeet



TDD это итеративный процесс



Test Last vs TDD

Test Last

- Не всегда есть время на тесты
- Тесты никто писать не любит
- Понимание реализации влияет на тесты
- Иногда получаются уродливые тесты

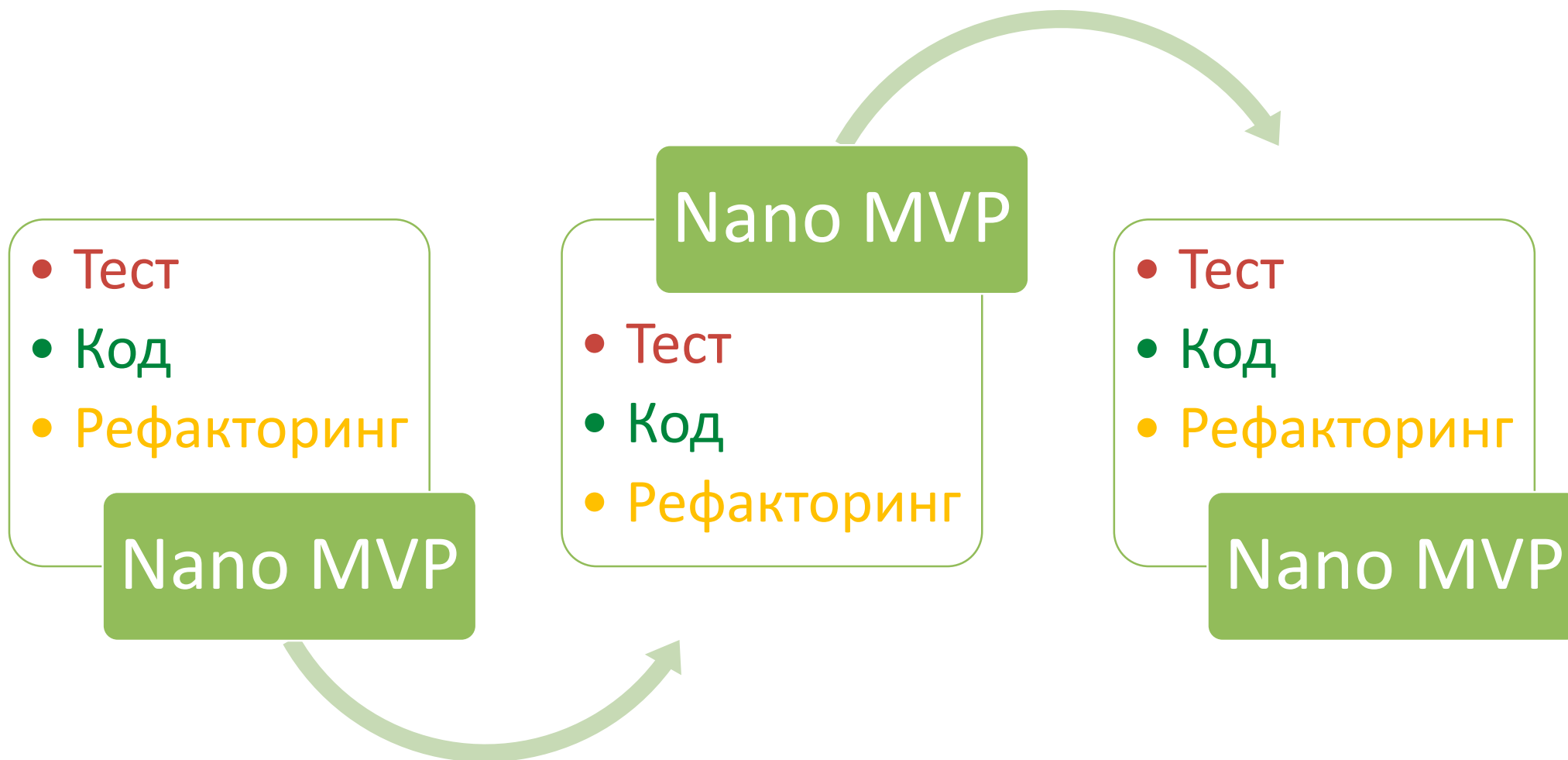
TDD

- **Всегда** есть тесты
- Тесты становится писать приятно
- Код косвенно **влияет** на тесты
- Иногда получаются уродливые тесты **в конце**



TDD vs Test First/Last –
как Agile vs Waterfall

TDD = Инкрементальное кодирование



TDD: Проблемы

- Высокий порог входа, особенно для опытных разработчиков
- «Рванный» ритм разработки, сложно войти в состояние потока
- Фундаментализм

TDD is dead. Long live testing.

“Test-first **fundamentalism** is like abstinence-only sex ed: An unrealistic, ineffective morality campaign for self-loathing and shaming.”

-David Heinemeier Hansson





Global Day of Coderetreat

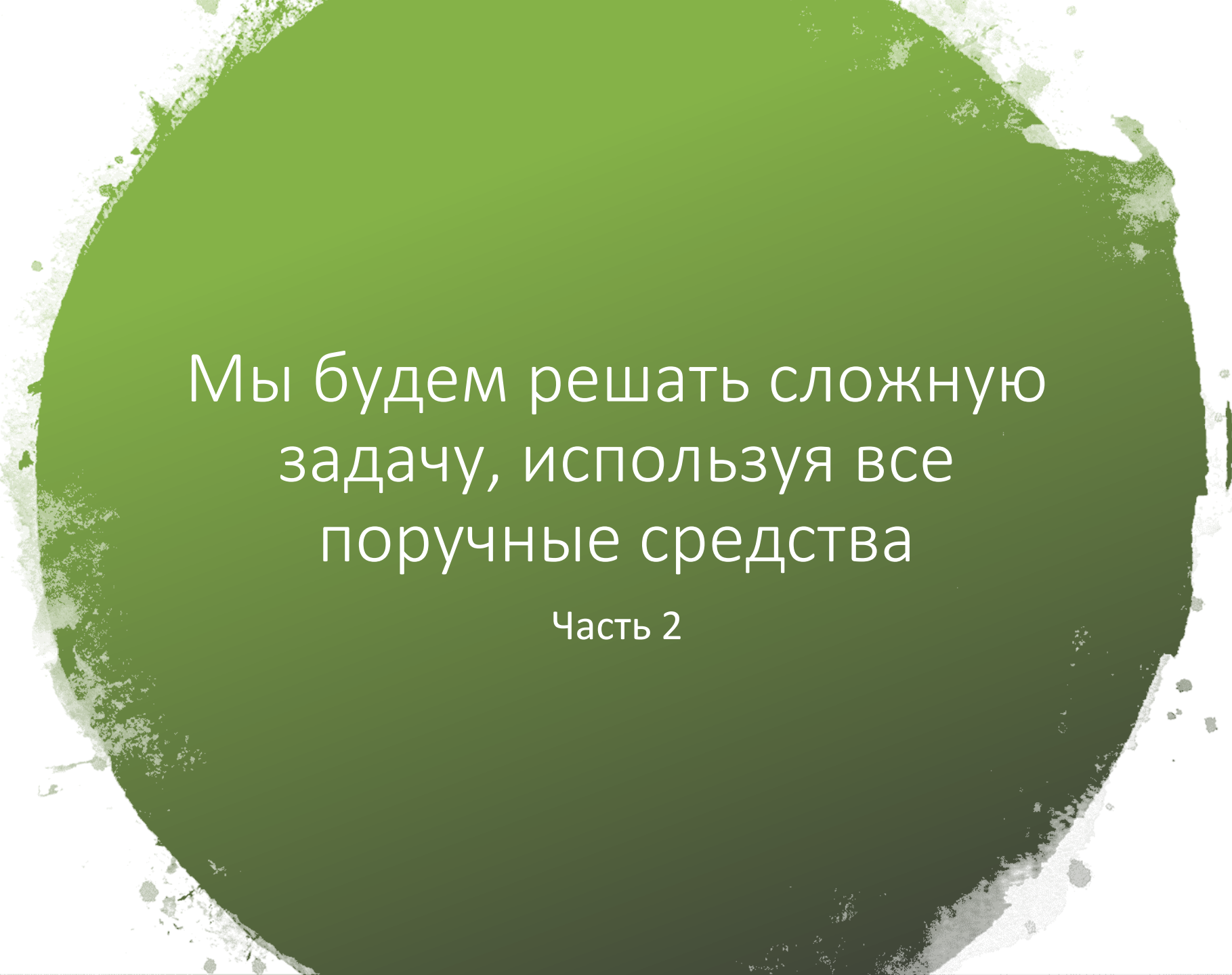
- Problem: Conway's Game of Life
- Length of Session: 45 minutes
- Duration: 8.30am to 5 or 6pm
- Pair-programming is necessary, as the knowledge transfer contained in that activity is essential to the practice
- Prefer using Test-Driven Development (TDD)
- After each session, pairs should be swapped
- After each session, code must be deleted, not put in a branch, not stashed, just deleted with no trace left

TDD: Цитата

*"Test-driven development is a way
of managing fear during
programming"*

- Kent Beck

TDD
Red Green
Refactoring



Мы будем решать сложную
задачу, используя все
поручные средства

Часть 2



Алексей Березин — популярнейший блогер из Сибири, на его ЖЖ «Слон в колесе» (<https://alex-aka-jj.livejournal.com/>) подписаны более 22 000 человек. По рассказу «Совещание» были сняты 4 короткометражных экранизации в России, Украине и Великобритании. Британская версия («The Expert») была переведена на 50 языков и к 2018 г. собрала около 20 000 000 просмотров на YouTube. Фильм «Профессионалы», снятый украинским коллективом Anre (реж. К. Шлямин) по мотивам рассказа «Совещание», получил диплом «За лучший сценарий» на IV Международном фестивале молодого кино (Киев, 2012).

Запомните: в русской литературе появился новый замечательный рассказчик. С сильным неповторимым голосом. Который не спутаешь ни с чьим другим...

Валерий Хаит,
главный редактор одесского юмористического журнала «Фонтан»

Юмор Алексея отчаянно галантен и слегка застенчив. Он ненавязчив, словно редкий и очень желанный гость, которого хочется приглашать ещё и ещё. А ещё... впрочем, не буду портить вам сюрприз.

Максим Малявин,
«Блог Добрых Психиатров»

Рассказы Алексея Березина, как глоток свежезаваренного кофе — дарят радость и заставляют улыбнуться. И этого так не хватает в современных реалиях — доброты и смеха, особенно когда это смех соседа по вагону метро, который едет с такой же книжкой.

Екатерина Безьянная,
блог «Разноцветная жизнь»

www.astrel-spb.ru

АЛЕКСЕЙ
БЕРЕЗИН



7 КРАСНЫХ ЛИНИЙ
ЖЖ «Слон в колесе»



Семь красных линий

Нам нужно нарисовать семь красных линий. Все они должны быть строго перпендикулярны, и кроме того, некоторые нужно нарисовать зеленым цветом, а еще некоторые — прозрачным, и одну линию изобразить в виде котенка.



Нарисовать ...

Нарисовать ...

Code

```
public Line[]  
{  
  
  
  
  
  
  
}
```

Test

```
[Test]  
public void TestDraw()  
{  
    // arrange  
    // act  
    // assert  
}
```





... 7 красных линий...

... 7 красных линий...

```
public Line[] Draw()
```

{

}

... 7 красных линий...

- 7 штук
- 7 линий
- 7 красных линий

... 7 красных линий...

[Test]

```
void TestDraw_Returns7Items() {  
    // arrange  
    var expert = new TheExpert();  
    // act  
    var items = expert.Draw();  
    // assert  
    Assert.AreEqual(7, items.Length);  
}
```



TDD
Red Green
Refactoring

... 7 красных линий..

[Test]
public void TestDraw_Returns7Items()

[Test]
public void TestDraw_Returns10Items()

[Test]
public void TestDraw_Returns15Lines()

...





...строго перпендикулярных...

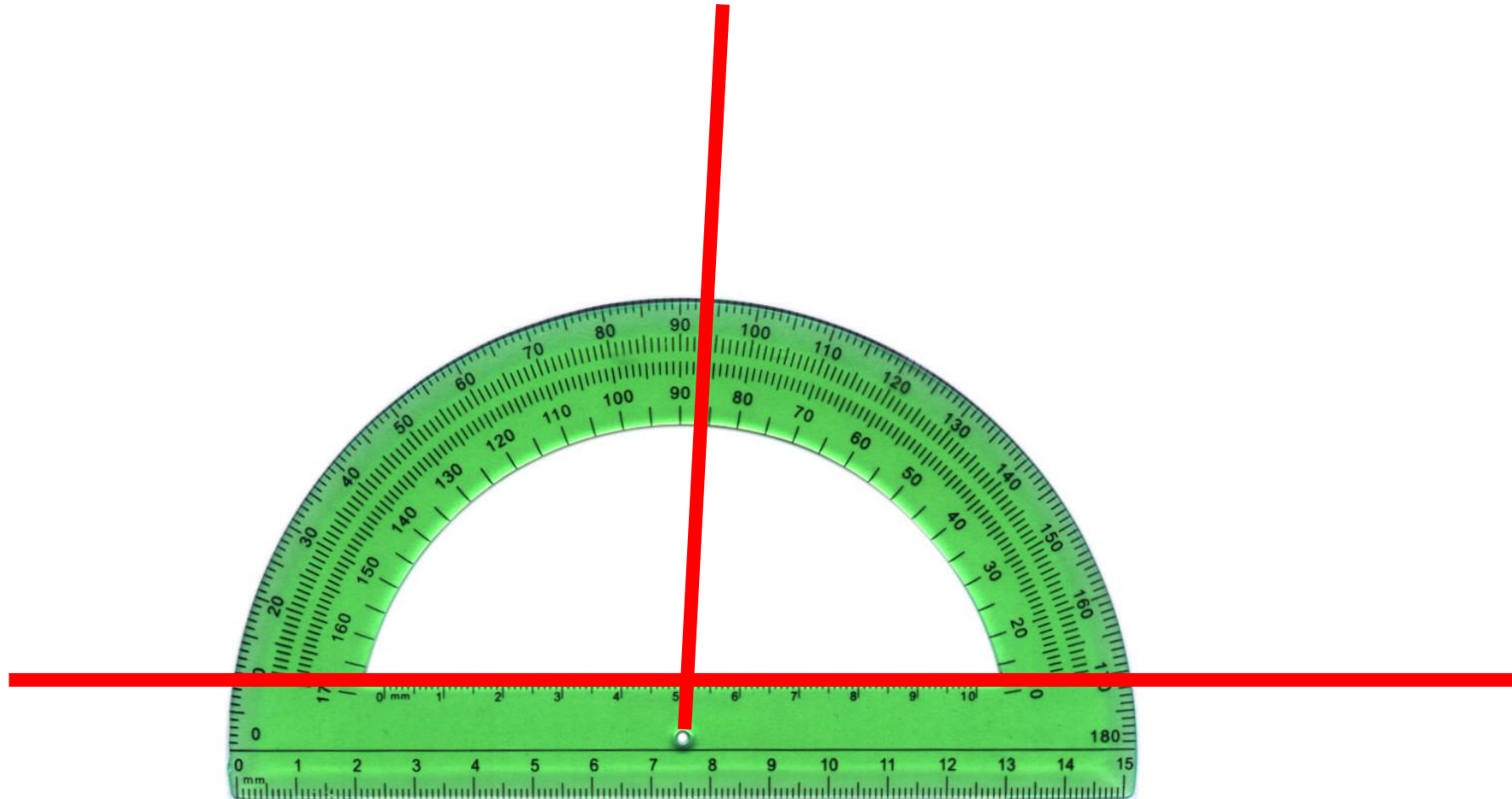
...строго перпендикулярных...

- 2 строго перпендикулярные друг другу линии
- 3 строго перпендикулярные друг другу линии
- 4 строго перпендикулярные друг другу линии
- 5 строго перпендикулярные друг другу линии
- 6 строго перпендикулярные друг другу линии
- 7 строго перпендикулярных друг другу линий

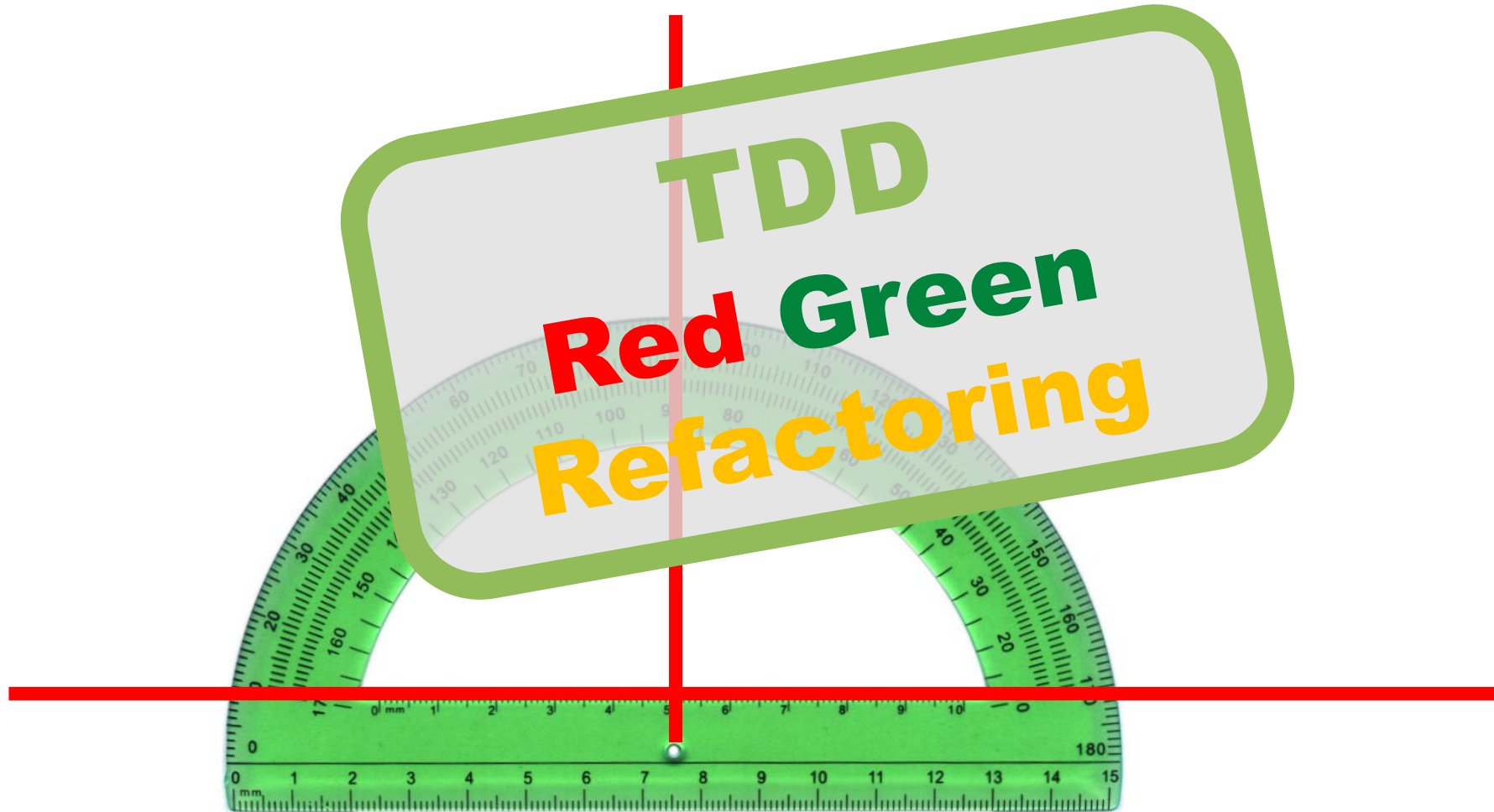


2 строго перпендикулярные линии

2 строго перпендикулярные линии



2 строго перпендикулярные линии



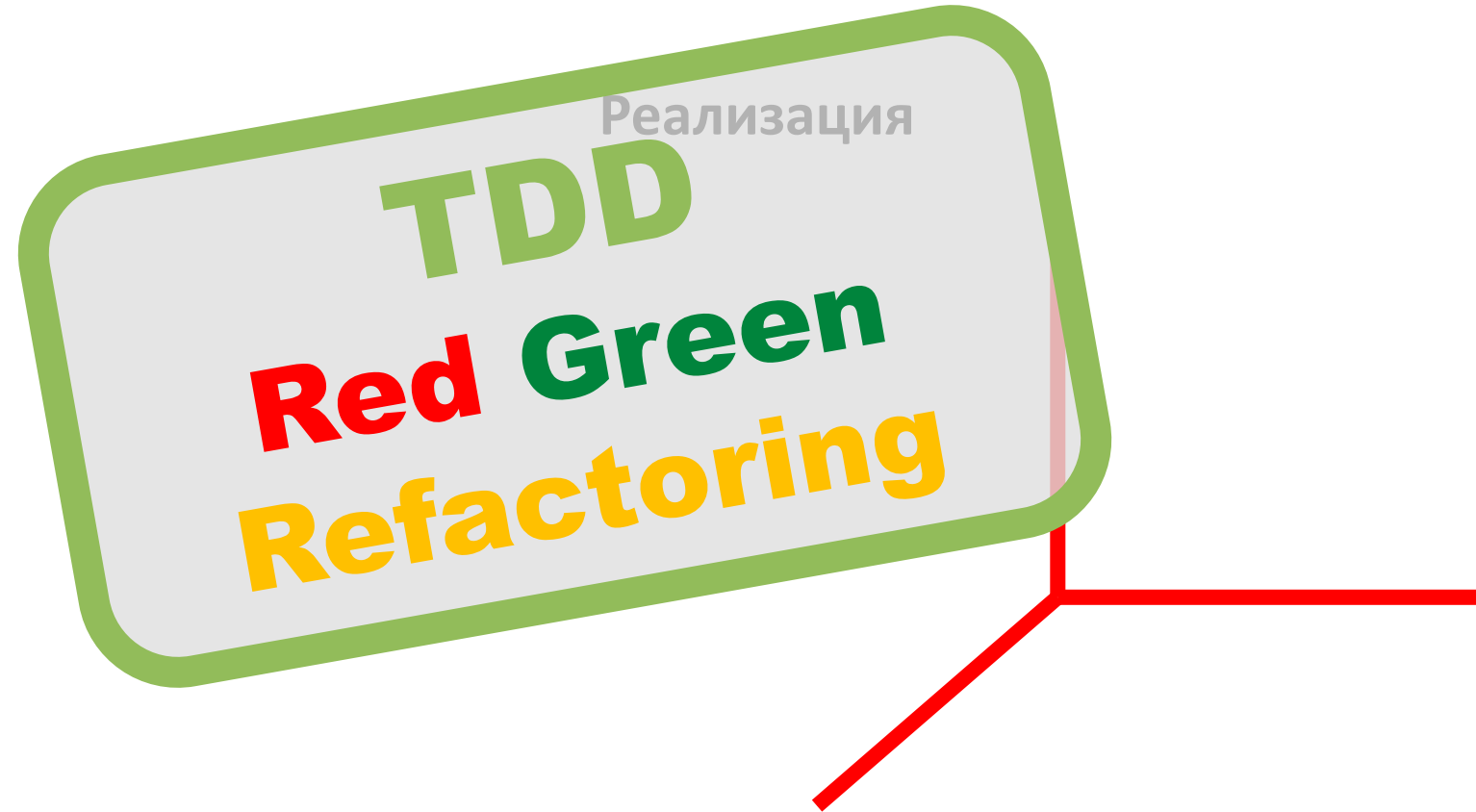
...строго перпендикулярных...

- ~~2 строго перпендикулярные друг другу линии~~
- 3 строго перпендикулярные друг другу линии
- 4 строго перпендикулярные друг другу линии
- 5 строго перпендикулярные друг другу линии
- 6 строго перпендикулярные друг другу линии
- 7 строго перпендикулярных друг другу линий

3 строго перпендикулярные линии

Тест

- line1 \perp line2
- line2 \perp line3
- line3 \perp line1



...строго перпендикулярных...

- ~~2 строго перпендикулярные друг другу линии~~
- ~~3 строго перпендикулярные друг другу линии~~
- 4 строго перпендикулярные друг другу линии
- 5 строго перпендикулярные друг другу линии
- 6 строго перпендикулярные друг другу линии
- 7 строго перпендикулярных друг другу линий

A portrait of a smiling man with dark hair, wearing a light blue button-down shirt. He is looking slightly to the right of the camera. The background is a soft, out-of-focus grey.

Состояние потока



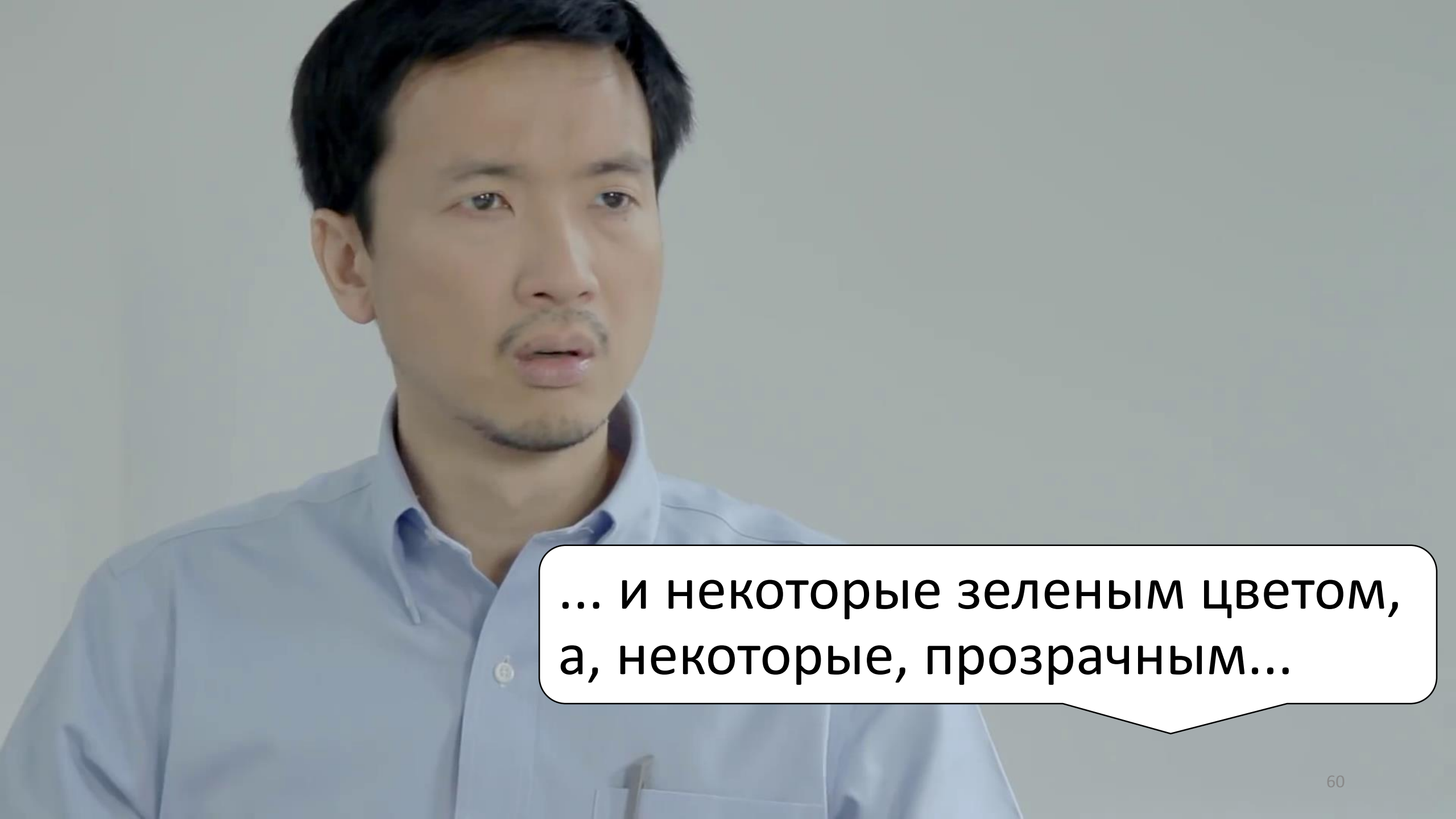
Global Day of Coderetreat

Они говорят,
состояние потока:

- Это плохо
- Не продуктивно
- Вредит качеству
- Не позволяет
писать красивый
код

Состояние потока — лучший мотиватор



A medium shot of a man with dark hair and a light beard, wearing a light blue button-down shirt. He is looking slightly to the right of the camera with a neutral expression. The background is a plain, light-colored wall.

... и некоторые зеленым цветом,
а, некоторые, прозрачным...

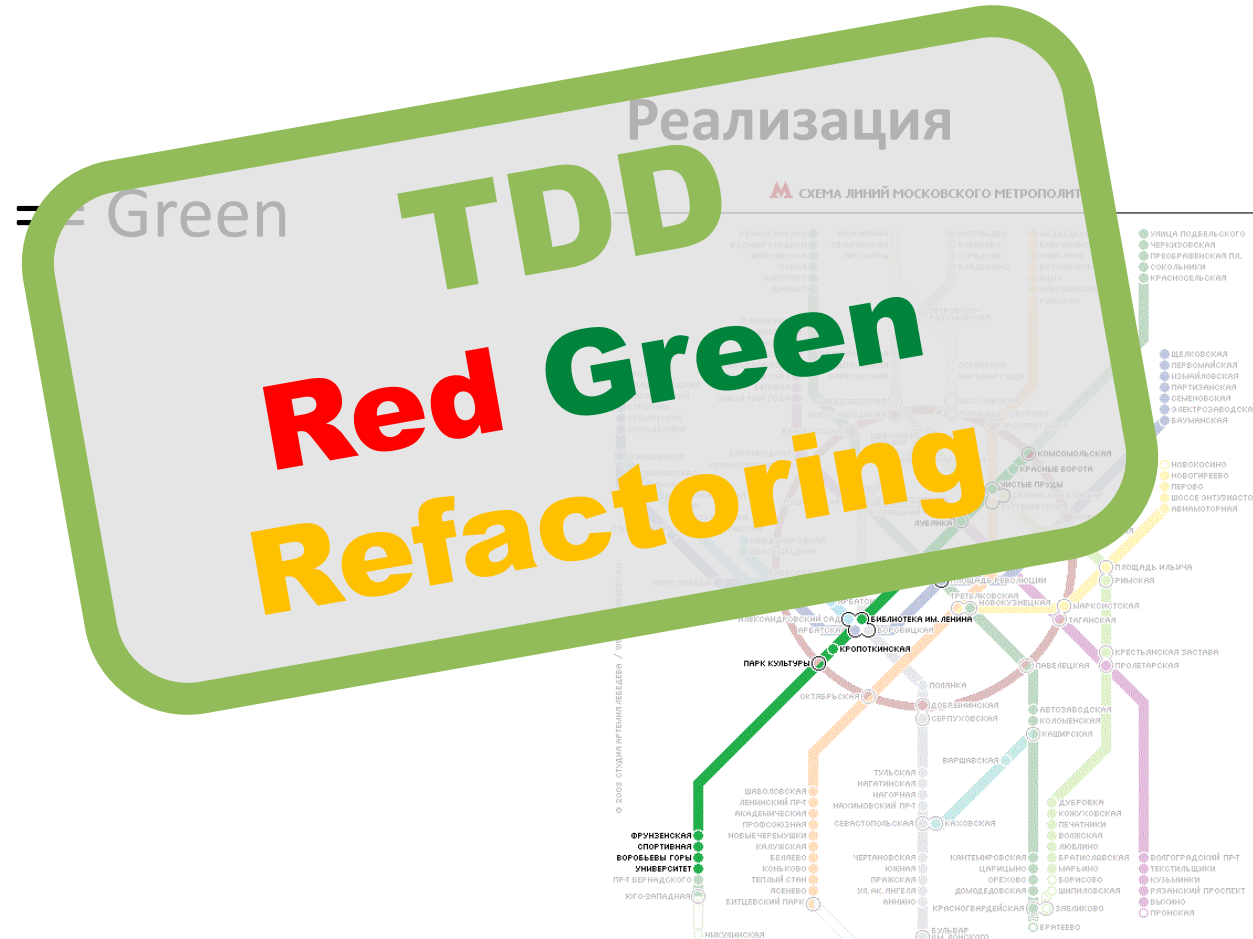
... и некоторые зеленым цветом, а, некоторые, прозрачным...

- Одна красная линия зеленым цветом
- Несколько красных линий зеленым цветом
- Одна красная линия прозрачным цветом
- Несколько красных линий прозрачным цветом

Одна красная линия зеленым цветом

Тест

- `redLine1.Color = Green`



... и некоторые зеленым цветом, а, некоторые, прозрачным...

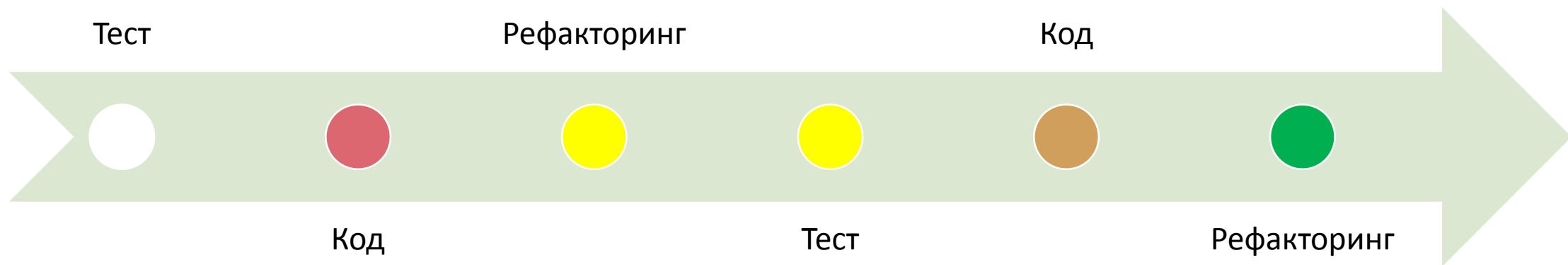
- ~~Одна красная линия зеленым цветом~~
- Несколько красных линий зеленым цветом
- Одна красная линия прозрачным цветом
- Несколько красных линий прозрачным цветом



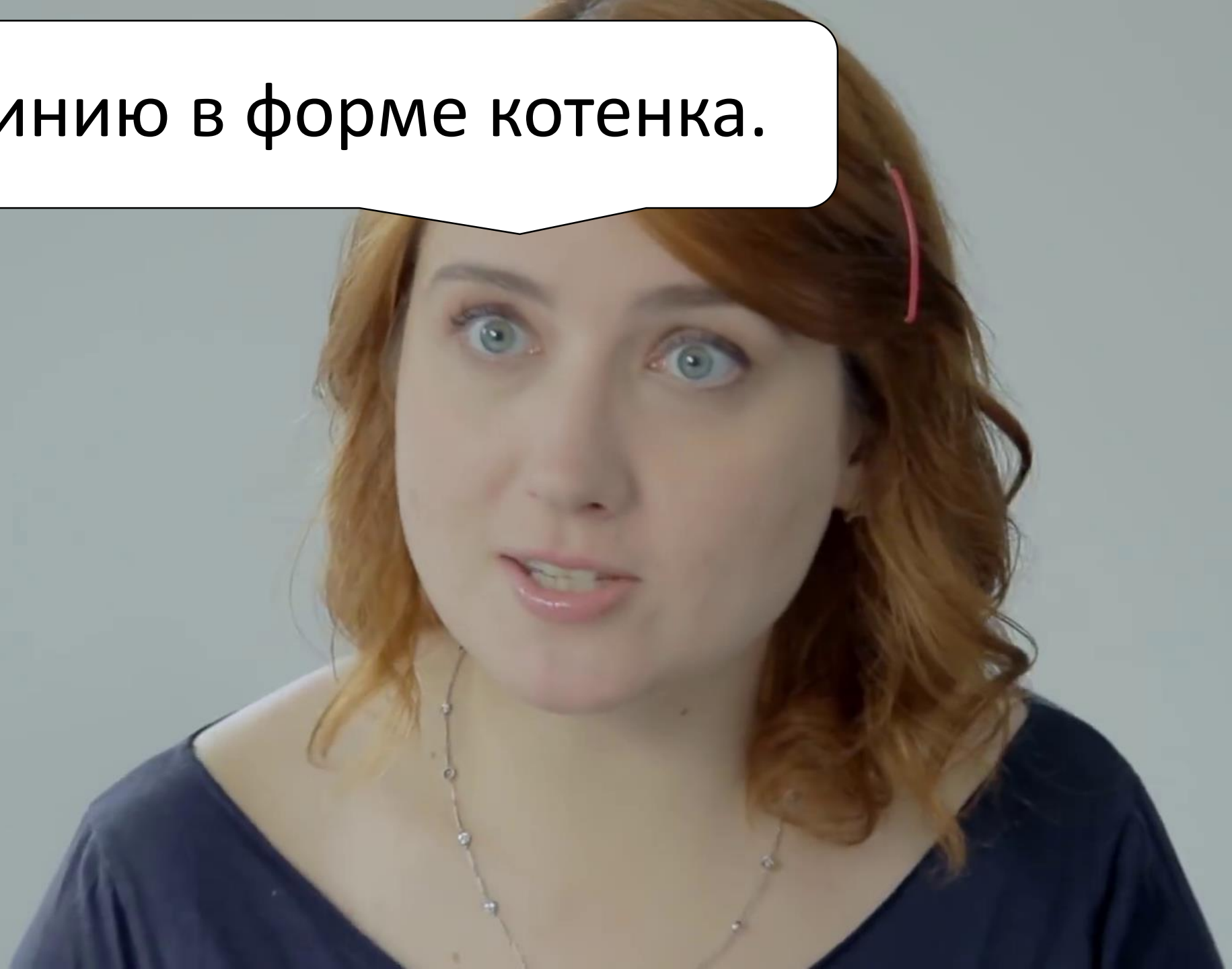
TDD
Red Green
Refactoring

Но ты же только что набыдлокодил

TDD, Быдлокод и Мотивация



...одну линию в форме котенка.



...Одну линию в форме котенка.

```
legacy.Legacy();  
var legacy1 = legacy.Legacy + legacyLegacy;  
legacy1.Legacy(le, ga, cy);  
var legacy42 = legacy  
    .Where(l => l.IsLeagacy)  
    .Select(l => l.Legacy);  
legacyService.MakeLegacy(legacy42);
```


...Одну линию в форме котенка.

```
legacy.Legacy();  
var legacy1 = legacy.Legacy + legacyLegacy;  
legacy1.Legacy(le, ga, cy);  
  
var legacy42 = legacy  
    .Where(l => l.IsLeagacy)  
    .Select(l => l.Legacy);  
legacyService.MakeLegacy(legacy42);
```

...Одну линию в форме котенка.

```
legacy.Legacy();  
var legacy1 = legacy.Legacy + legacyLegacy;  
legacy1.Legacy(le, ga, cy);  
legacy.Shape = legacy.Shape.ToKitty();  
var legacy42 = legacy  
    .Where(l => l.IsLeagacy)  
    .Select(l => l.Legacy);  
legacyService.MakeLegacy(legacy42);
```

...одну линию в форме котенка.



...Одну линию в форме котенка.

```
legacy.Legacy();
```

```
var legacy1 = legacy.Legacy + legacyLegacy;
```

```
legacy1.Legacy(le, ga, cy)
```

NullPointerException

```
legacy.Shape = legacy.Shape.Tokitsy(),
```

```
var legacy42 = legacy
```

```
    .Where(l => l.IsLeagacy)
```

```
    .Select(l => l.Legacy);
```

```
legacyService.MakeLegacy(legacy42);
```

...Одну линию в форме котенка.

```
legacy.Legacy();  
var legacy1 = legacy.Legacy + legacyLegacy;  
legacy1.Legacy(le, ga, cy);  
//legacy.Shape = legacy.Shape.ToKitty();  
var legacy42 = legacy  
    .Where(l => l.IsLeagacy)  
    .Select(l => l.Legacy);  
legacyService.MakeLegacy(legacy42);
```

...Одну линию в форме котенка.

```
[Test]  
public void Draw_LineIsLast_ReturnsKitty();
```

```
[Test]  
public void Draw_NoShape_DoNothing();
```

...

...Одну линию в форме котенка.

```
legacy.Legacy();  
var legacy1 = legacy.Legacy + legacyLegacy;  
legacy1.Legacy(le, ga, cy);  
legacy.Shape = legacy.Shape?.ToKitty();  
var legacy42 = legacy  
    .Where(l => l.IsLeagacy)  
    .Select(l => l.Legacy);  
legacyService.MakeLegacy(legacy42);
```


...одну линию в форме котенка.



Offtopic

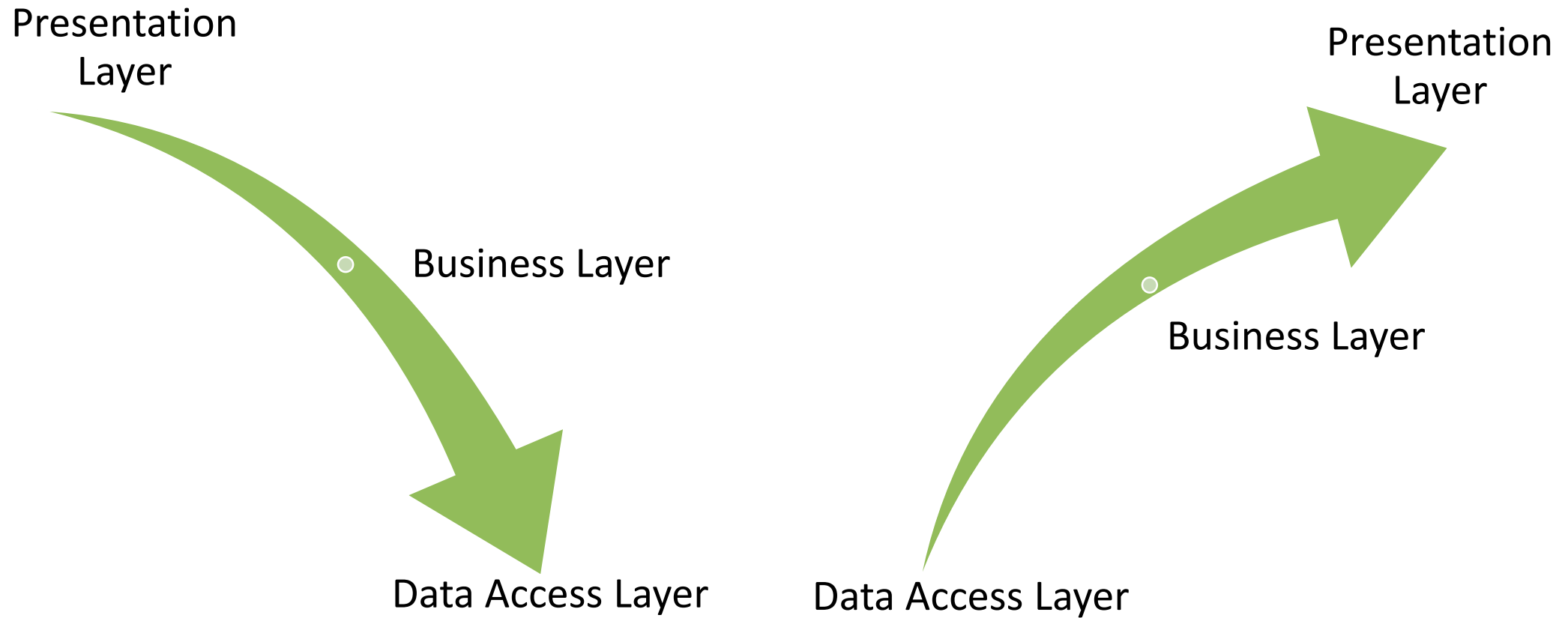
Часть 3

Offtopic №1: Готовое решение

- GOF паттерны
- Domain паттерн
- Copy-Paste с прошлого проекта
- Очень простая задача



Offtopic №2: Отладка Data Access Layer

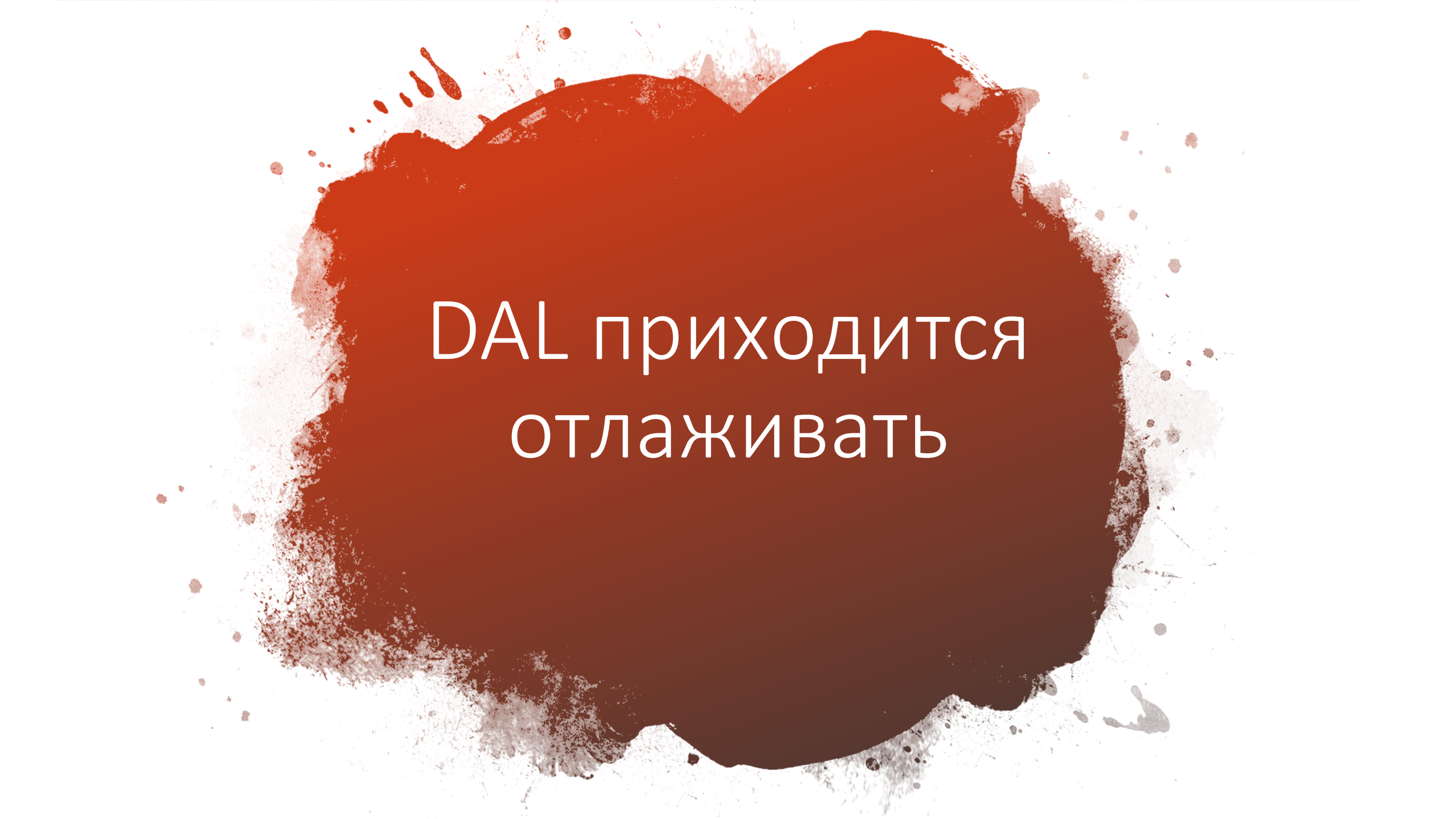


Data Access Layer: Entity Framework

```
var v = from m in WebAppDbContext.UserSessionTokens
        from c in WebAppDbContext.Companies.Include(a => a.SecurityGroups)
        from n in WebAppDbContext.SecurityGroups.Include(x => x.Members)
        where m.TokenString == userTokenString &&
               n.Members.Contains(m.User) &&
               c.SecurityGroups.Contains(n)
        select c;
```

Data Access Layer: Elasticsearch

```
var searchResults = searchClient.Client.Search<Submitter>(x => x
    .Index(clientIndexName)
    .Type(_elasticConfiguration.UserInfoTypeName)
    .From(query.PagingInfo.StartRow)
    .Size(query.PagingInfo.PageSize + 1)
    .Sort(s => s.Ascending(f => f.UserName.Suffix(KeywordExt)))
    .Query(
        q => q.Bool(
            b => b.MustNot(
                m => m.Term(t => t.Field(f => f.IsGlobalAdmin).Value(true)),
                m => m.Term(t => t.Field(f => f.IsClientAdmin).Value(true)),
                m => m.Term(t => t.Field(f => f.IsAllClientsAdmin).Value(true))
            ).Filter(f => f.Bool(
                innerbool => innerbool.Should(
                    s => s.Match(m => m.Query(query.Search).Field(innerf => innerf.UserName).Operator(Operator.And)),
                    s => s.Match(m => m.Query(query.Search).Field(innerf => innerf.EmailAddress).Operator(Operator.And)))
            )))
    ));
```

A large, irregular red ink splatter or blotch serves as the background for the text. The splatter is centered and has a textured, painterly appearance with various shades of red and some darker spots. The text is white and centered within the red area.

DAL приходится
отлаживать

Как мы обычно отлаживаем?



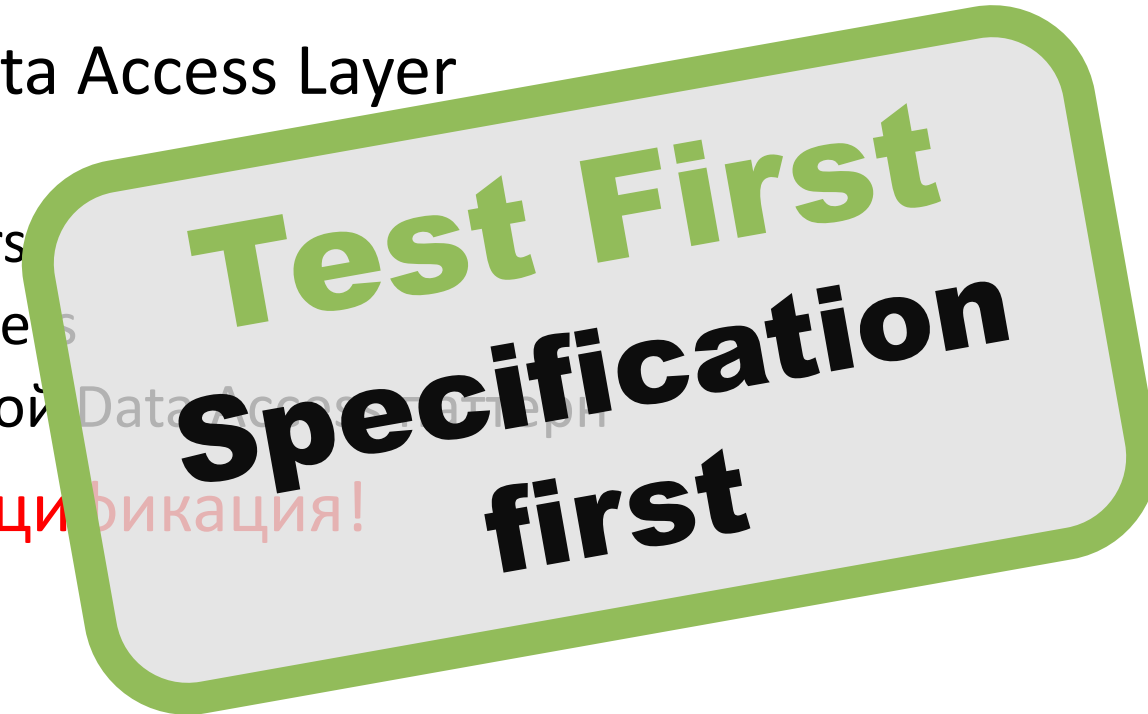
- Запустить:
 - 0.2-5 минут
- Найти страницу:
 - 0-2 минуты
- Воспроизведение:
 - 1-10 минут
- Получить результат:
 - 0-10 минут
- Анализ:
 - ...

Что же делать?

- Интерфейс Data Access Layer

- Repositories
- Data Providers
- Ports – Adapters
- Вообще любой Data Access Layer

- У нас есть спецификация!



Что нам необходимо?

- Инфраструктура
 - Docker
 - Powershell и засучить рукава
 - Забить на CI и **обеспечить запускаяемость у разработчиков**
- Тестовые данные
- ???
- PROFIT

Интеграционные Тесты Dal

```
[Test]
void Find_EmptySearch_ReturnsAll();

[Test]
void Find_1WordSearch_ReturnsStartsWith();

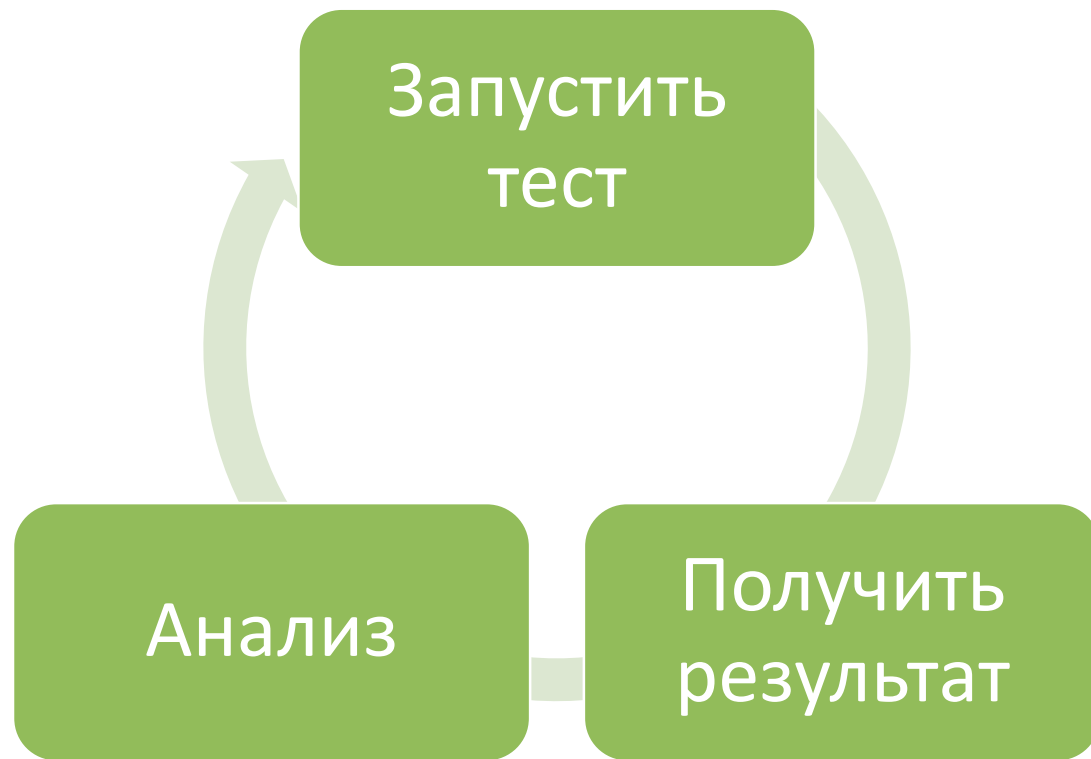
[Test]
void Find_2WordsSearch_ReturnsByNGram();

[Test]
void Find_SearchNotExists_ReturnsEmpty();

[Test]
void Find_ExistsButNoPermissions_ReturnsEmpty();
```

```
interface EntityRepository
{
    IEnumerable<Entity> Find(string search);
}
```

Как теперь отлаживать?



- Запустить и получить результат :
 - 0.2-20 минут
- Анализ:
 - ...

Заключение

Давайте использовать все инструменты

- Test First
 - Когда Вы знаете что от вас требуется



Давайте использовать все инструменты

- Test First
 - Когда Вы знаете что от вас требуется
- TDD
 - Когда мы не можем сделать первый шаг
 - Когда мы демотивированны
 - Когда хочется побыдлокодить



Давайте использовать все инструменты

- Test First
 - Когда Вы знаете что от вас требуется
- TDD
 - Когда мы не можем сделать первый шаг
 - Когда мы демотивированны
 - Когда хочется побыдлокодить
- Test Last
 - Когда пришло вдохновение
 - Когда знаете как делать



Полезные ресурсы

- Экстремальное программирование: разработка через тестирование – Кент Бек
- Эффективная работа с унаследованным кодом – Майкл К. Физерс
- TDD is dead. Long live testing – David Heinemeier Hansson: david.heinemeierhansson.com/2014/tdd-is-dead-long-live-testing.html
- Global Day of Coderetreat: www.coderetreat.org
- INTRODUCING BDD: dannorth.net/introducing-bdd

Спасибо за ВНИМАНИЕ

Вопросы?

kugushew@gmail.com