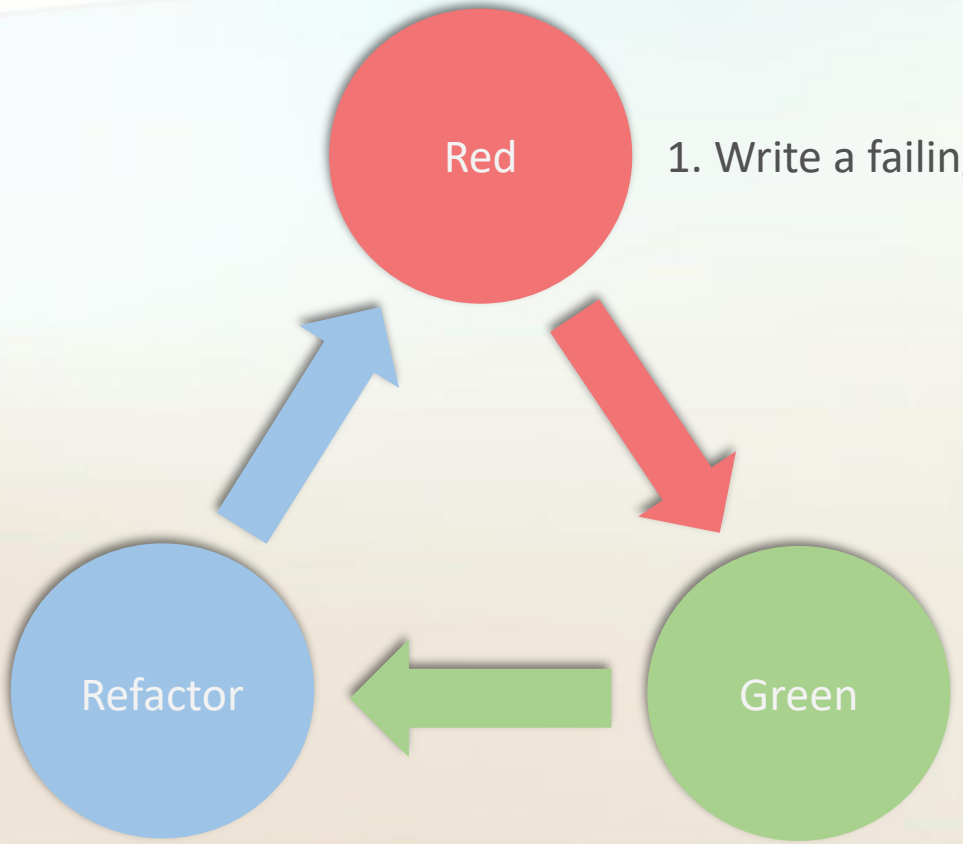
XLENT
CONSULTING GROUP

Test Driven Development – TDD

Peter Starefeldt

XLENT
CONSULTING GROUP

What is Test Driven Development?



1. Write a failing test

2. Make the test pass

3. Optimize the code's design

XLENT
CONSULTING GROUP

Starting with the test

```
public void AddCourse(Course course);
```

XLENT
CONSULTING GROUP

Starting with the test

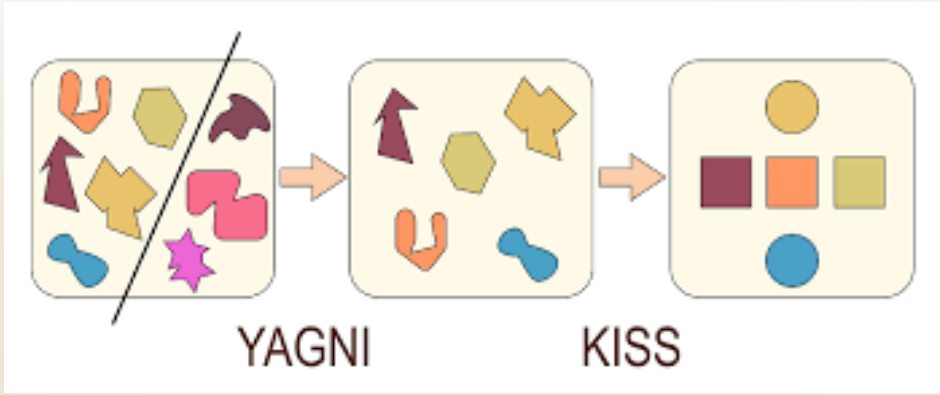
```
public void AddCourse(Course course)
{
      var errors = course.IsValid();

       If(errors.Any())
       {
              _repository.AddValidationErrors(errors);
       }
       else
       {
              _repository.Add(course);
       }
}
```

XLENT
CONSULTING GROUP

Having a failing test

- Build exactly what is needed – nothing more!
- Unnecessary features – big reason for system failures.
- YAGNI – You Aren't Gonna Need It
- KISS – Keep It Simple, Stupid!



XLENT
CONSULTING GROUP

Make it Pass!

```
public void Validate_WhenAgeIsUnder18_ReturnsFalse()
{
        //Arrange
        var validator = new StudentValidator();
        var student = new Student();
        student.Age = 17;

        //Act
        var result = validator.Validate(student);

        //Assert
        Assert.IsFalse(result);
}
```

XLENT
CONSULTING GROUP

Make it Pass!

```
public void Validate_WhenAgeIsUnder18_ReturnsFalse()
{
        //Arrange
        var validator = new StudentValidator();
        var student = new Student();
        student.Age = 17;

        //Act
        var result = validator.Validate(student);

        //Assert
        Assert.IsFalse(result);
}
```

```
public bool Validate(Student student)
{
    return false;
}
```

XLENT
CONSULTING GROUP

Make it Pass!

```
public void Validate_WhenAgeIsOver17_ReturnsTrue()
{
        //Arrange
        var validator = new StudentValidator();
        var student = new Student();
        student.Age = 18;

        //Act
        var result = validator.Validate(student);

        //Assert
        Assert.IsTrue(result);
}
```

XLENT
CONSULTING GROUP

Make it Pass!

```
public void Validate_WhenAgeIsOver17_ReturnsTrue()
{
        //Arrange
        var validator = new StudentValidator();
        var student = new Student();
        student.Age = 18;

        //Act
        var result = validator.Validate(student);

        //Assert
        Assert.IsTrue(result);
}
```

```
public bool Validate(Student student)
{
    return student.Age > 17;
}
```

XLENT
CONSULTING GROUP

Refactoring

- Change the code without changing its behavior.
- For example:
  - Breaking out a part of code from one method into a new method.
  - Breaking out new classes from one big class.
- Remove duplicated code.
- Duplicated code becomes a problem when changes occur.

XLENT
CONSULTING GROUP

Word of advice

- **Best when:**
  - Expected behavior is known.
  - Lots of use cases/ scenarios.
  - Team has a similar drive to use it.
  - Project is complex and involves lots of future maintenance/new features.
  - Customers are looking for cost reductions long term.

- **More difficult to use when:**
  - Requirements are not really known.
  - Experimenting with code design.
  - Developer is inexperienced.

XLENT
CONSULTING GROUP

This presentation with exercise-material can be found at:

- https://github.com/starefeldt/presentation-unittest-tdd

XLENT
CONSULTING GROUP

Resources:

- *Test Driven Development, By Example* – Kent Beck (2003)
- *Working Effectively with Legacy Code* – Michael Feathers (2004)
- *Clean Code* – Robert C Martin (2008)
- *The Art of Unit Testing* – Roy Osherove (2014)
- *Dependency Injection Principles, Practices, and Patterns* – Steven van Deursen and Mark Seemann (2019)
- *Pluralsight* – online learning platform