

```

        $course->students()->attach($studentId);
    }
}
}

```

It looks okay, but can you spot the problems?

- Arguments like these: array \$data or array \$lessons
- Lines like this: \$data['lessons']

My biggest problem is that **I don't want to maintain and debug massive associative arrays five years later.**

The above example is very basic. Now, please imagine your favorite legacy project, where you have to work with methods like this:

```

public function createProduct(array $data)
{
    // Insert 673 lines of code here

    /**
     * You have to reverse-engineer this whole shit-show
     * just to get an idea about the shape of $data, right?
     */
}

```

DTOs can solve this problem by structuring your unstructured data. The same CourseService with DTOs: