

Using Value Objects as Domain Model Building Blocks

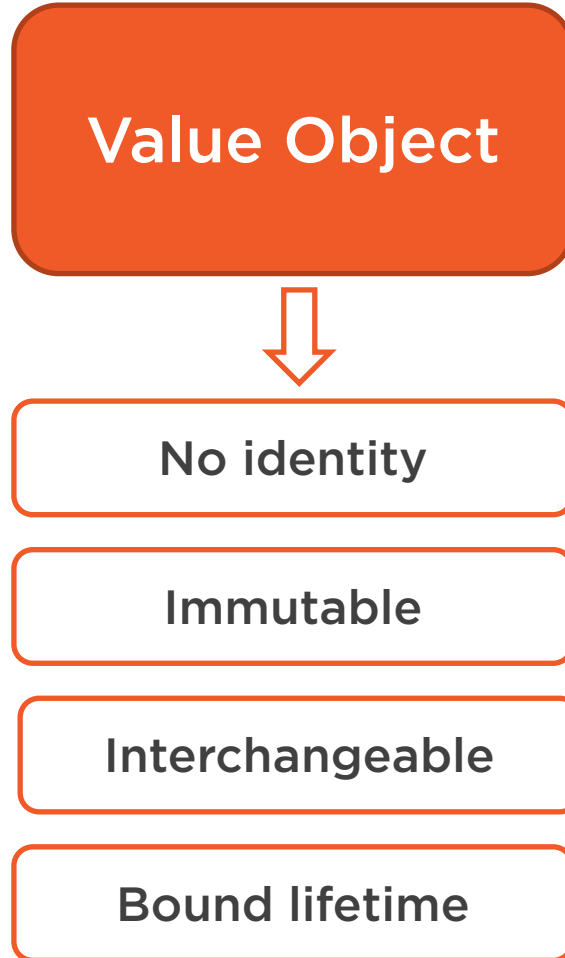


Vladimir Khorikov

@vkhorikov www.enterprisecraftsmanship.com



Importance of Using Value Objects



Domain-Driven Design in Practice

by Vladimir Khorikov

A descriptive, in-depth walk-through for applying Domain-Driven Design principles in practice.

▶ Resume Course

Table of contents

Description

Transcript

Exercise files

Discussion

Learning Check

Recommended



Introduction



29m 31s



Starting with the First Bounded Context



46m 18s

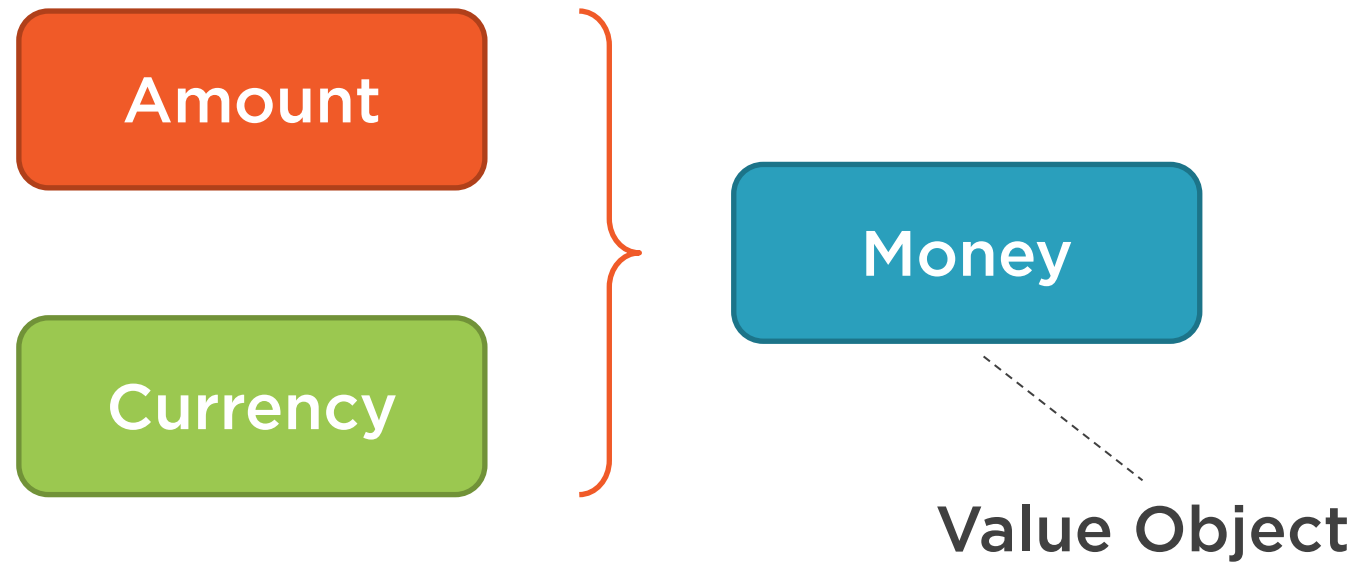


Introducing UI and Persistence Layers



33m 20s

Importance of Using Value Objects

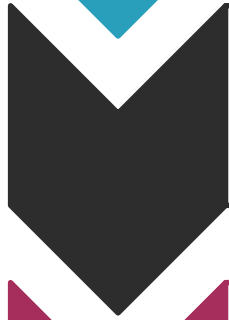


Single value is good too

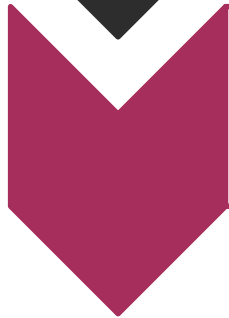
Importance of Using Value Objects



Identify problems faster, during compilation



More expressive domain model



Avoid domain logic duplication



Recap: Identifying Value Objects

- ✓ Identified three domain concepts
- ✓ Extracted two of them into Value Objects
- ✓ Fully utilize the type system

~~customer.Email = customer.Name~~



Applying Functional Principles in C#

by Vladimir Khorikov

Functional programming in C# can give you insight into how your programs will behave. You'll learn the fundamental principles that lie at the foundation of functional programming, why they're important, and how to apply them.

▶ Resume Course

Table of contents

Description

Transcript

Exercise files

Discussion

Recommended

▶ Course Overview



1m 15s

▶ Introduction



10m 49s

▶ Refactoring to an Immutable Architecture

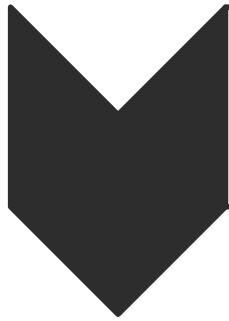


34m 53s

Recap: Gathering Validation Rules into a Single Place



Value Objects are fully encapsulated
Can't create invalid instances



No domain knowledge duplication

```
[MaxLength(100, ErrorMessage = "Name is too long")]  
public virtual string Name { get; set; }]
```



Recap: Introducing Dollars Value Object



Made the domain model more readable

```
Dollars  
private decimal CalculatePrice(/* ... */)
```



Additional factory method

```
Dollars price = Dollars.Of(4);  
Dollars price = (Dollars)4;
```



Summary



Using value objects as building blocks

- Raise the level of abstraction in code
- Make domain concepts explicit
- Gather related domain knowledge into a single place
- Fully utilize programming language's type system



In the Next Module

Pushing Logic Down from Services to Domain Classes

