

A decorative graphic on the left side of the slide. It consists of a blue parallelogram and a light green parallelogram, both tilted at an angle. The blue shape is in the foreground, and the green shape is partially behind it. They are set against a dark blue background with subtle diagonal lines.

# Design Patterns

by Versus 2020

# **Data Mapper**

## **Преобразователь данных**

Структурные шаблон





# Суть шаблона

Посредник для двунаправленной передачи данных между постоянным хранилищем данных и представления данных в памяти.



# Проблема

Объектно-реляционный разрыв - общая проблема  
любого языка ООП.



## Решение

Держать представление данных в памяти и постоянное хранилище данных независимыми друг от друга и от самого преобразователя данных.

# Структура

c	User
f	email
f	username
m	__construct(username, email)
m	getUsername()
m	fromState(state)
m	getEmail()

c	UserMapper
f	adapter
m	__construct(storage)
m	mapRowToUser(row)
m	findById(id)

c	StorageAdapter
f	data
m	__construct(data)
m	find(id)

# Пример кода





# Применимость

Если база данных с недружелюбной структурой, шаблон позволяет вам скрыть это.





# Шаги реализации

1. Создайте маппер объекта, передав связь адаптера хранилища.
2. Реализуйте метод получения свойств из хранилища (постоянное) и преобразование в объект (временное).
3. Для преобразования реализуйте методы маппинга и классе объекта.



# Преимущества

Позволяет работать со слоем данных как с объектами, при этом структура хранения данных в постоянном хранилище полностью инкапсулирована и может иметь любую архитектуру.

Каждый объект имеет свою зону ответственности, тем самым следуя принципам SOLID и сохраняя каждый объект простым и по существу.

Бизнес-логика и сохранение данных связаны слабо.



# Недостатки

Вам придется гораздо больше думать, перед тем как написать код.

В итоге у вас больше объектов в управлении, что немного усложняет код и его отладку.

**ИТОГ**

