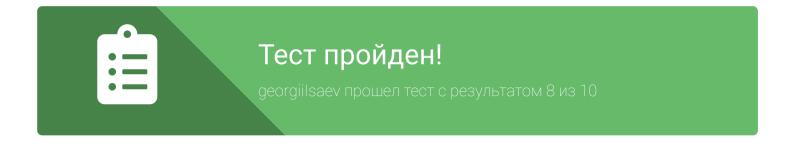
## ПАТТЕРНЫ ПРОЕКТИРОВАНИЯ 🔗



## Как вам тест?



## РЕЗУЛЬТАТЫ

32

Заработанные очки

junior Сложность 80%

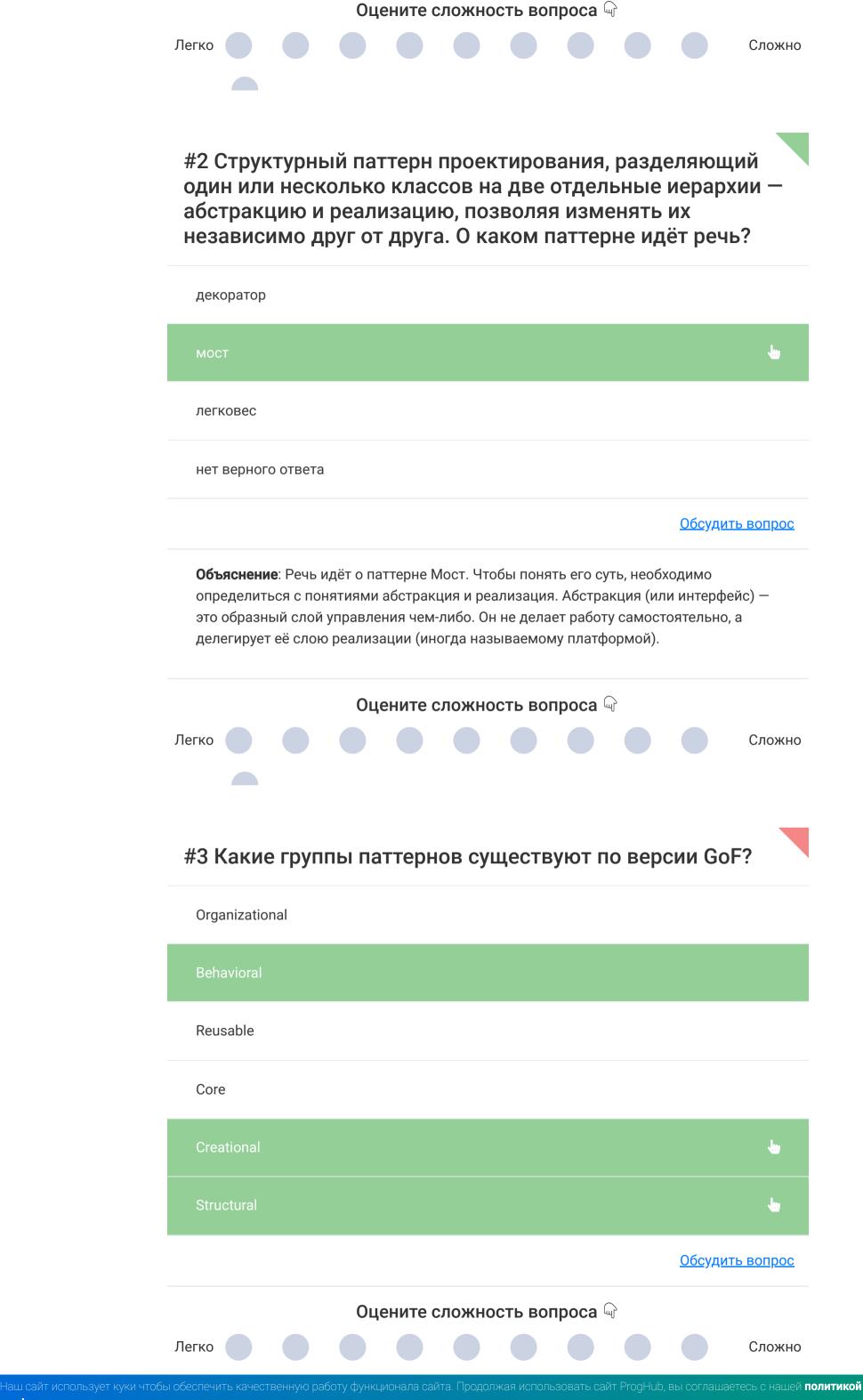
## **ОТВЕТЫ**

#1 Паттерн проектирования ... обеспечивает существование одного экземпляра некоторого класса и предоставляет единую точку доступа к нему.

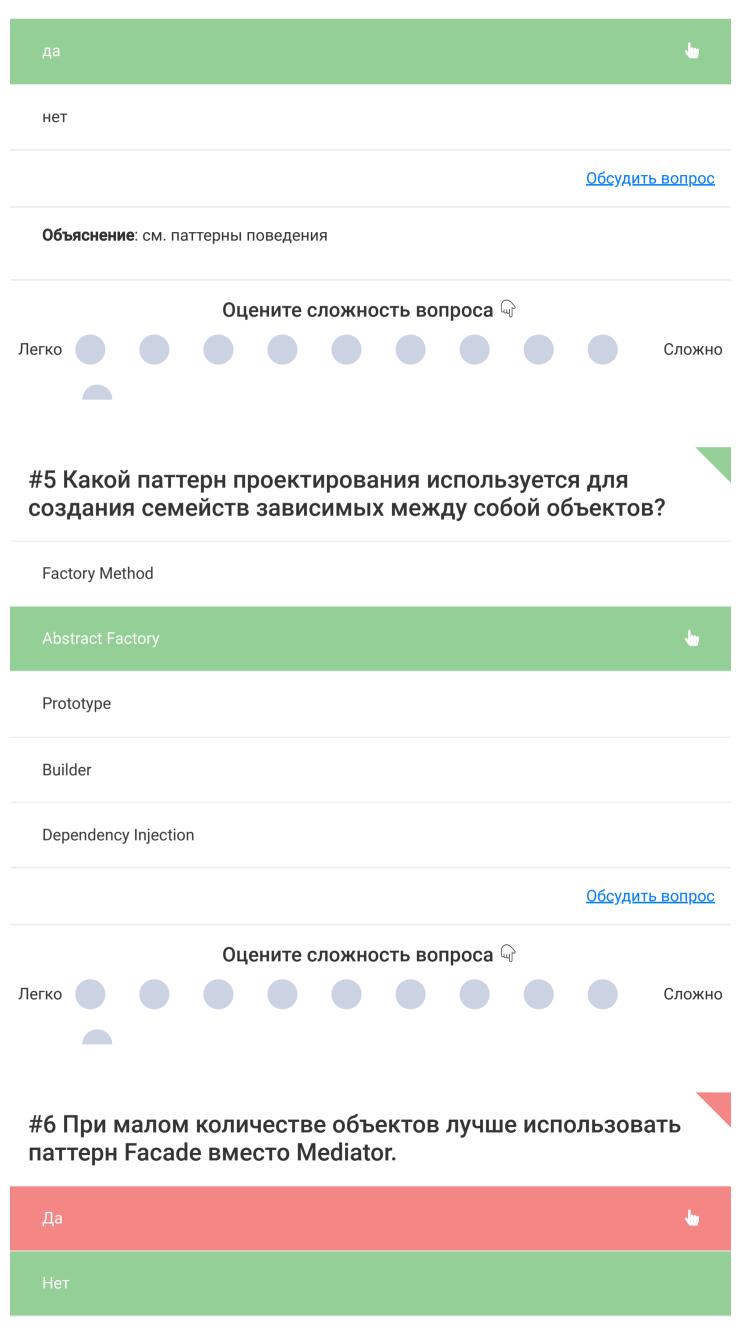


**Объяснение**: Паттерн Одиночка (англ. Singleton) — это порождающий паттерн проектирования, гарантирующий, что у класса есть только один экземпляр, и предоставляющий к нему глобальную точку доступа. Данный паттерн решает сразу две проблемы, нарушая принцип единственной ответственности класса:

- Гарантирует наличие единственного экземпляра класса. Чаще всего это полезно для доступа к какому-то общему ресурсу, например, базе данных.
- Предоставляет глобальную точку доступа. Это не просто глобальная переменная, через которую можно достучаться к определённому объекту. Глобальные переменные не защищены от записи, поэтому любой код может подменять их значения без вашего ведома.



#4 Верно ли высказывание: Различие между паттернами посредник (Mediator) и наблюдатель (Observer) в том, что наблюдатель распределяет обмен информацией между объектами, а посредник наоборот, инкапсулирует взаимодействие между другими объектами.

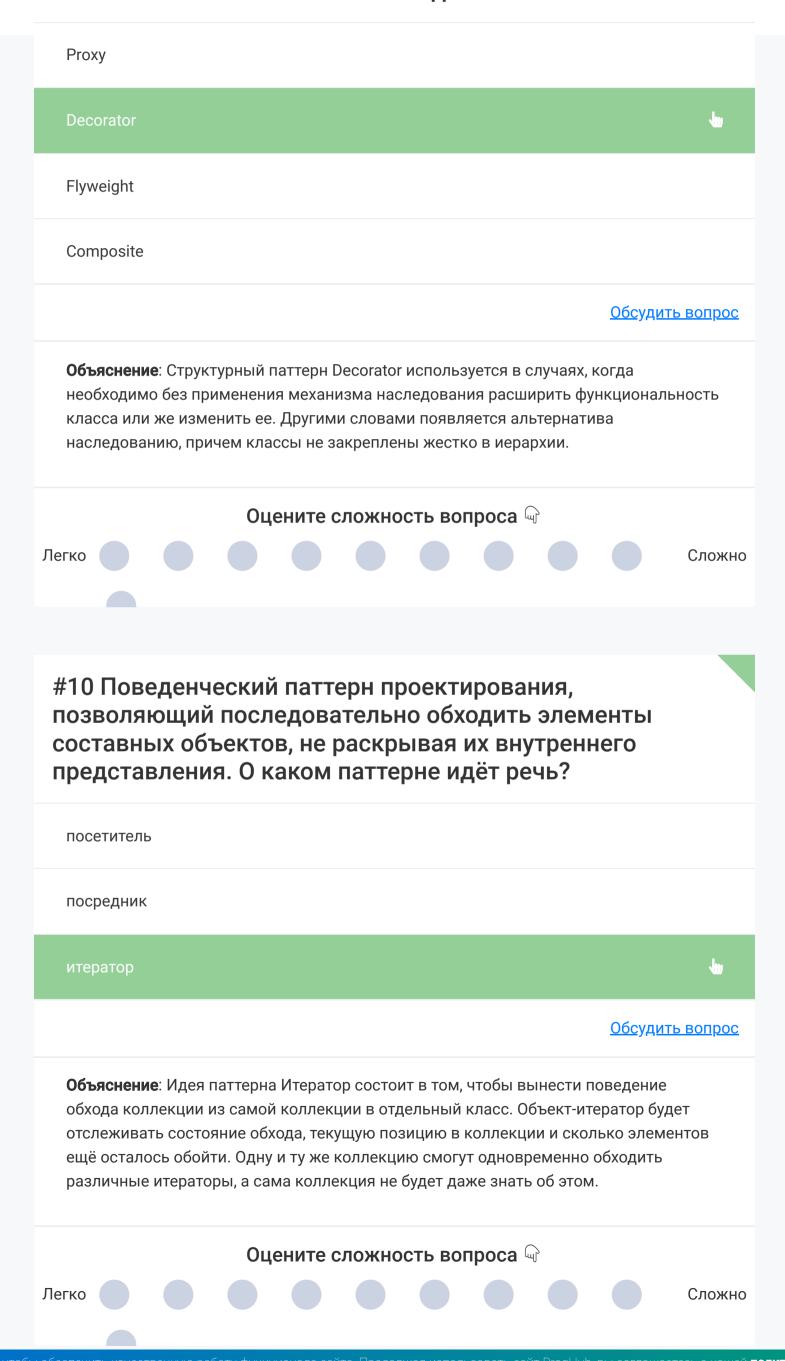


Обсудить вопрос

Обсудить вопрос

**Объяснение**: Паттерны проектирования — это многократно применяемая архитектурная конструкция, предоставляющая решение общей проблемы проектирования в рамках конкретного контекста и описывающая значимость этого решения. Паттерн не является законченным образцом проекта, который может быть прямо преобразован в код (что отличает его от близкого понятия идиомы), скорее это описание или образец для того, как решить задачу, таким образом, чтобы это можно было использовать в различных ситуациях. Объектно-ориентированные паттерны зачастую показывают отношения и взаимодействия между классами или объектами, без определения того, какие конечные классы или объекты приложения будут использоваться.

Главная польза каждого отдельного паттерна состоит в том, что он описывает решение целого класса абстрактных проблем. Также тот факт, что каждый паттерн имеет свое имя, облегчает дискуссию об абстрактных структурах данных (ADT) между разработчиками, так как они могут ссылаться на известные шаблоны. Таким образом, за счёт паттернов производится унификация терминологии, названий модулей и элементов проекта.





support@mg.proghub.ru