



Топ контрибуторов

За последние сутки

Нет данных

Получение Q



Знаете ли Вы, что

В разделе "Статьи" можно найти обучающие статьи по информационным технологиям, а также узнать о новостях сервиса Quizful.



Опросы

Какая область из перечисленных Вам наиболее интересна?

Мобильные технологии

Тестирование ПО

Защита информации

SEO / Интернет маркетинг

Компьютерные сети

Ответить

Лента обновлений

Jul 23 01:57

ссылка

Комментарий от

GeorgeMayoh:

Вообще я слышал такую версию: Если класс просто соде...

Jul 22 21:30

ссылка

Комментарий от leonx7:

Красиво подловил)

Jul 22 21:09

ссылка

Комментарий от leonx7:

Вопрос хороший, но точно не для теста, где время ограниче...

Jul 21 20:08

ссылка

Добавлен вопрос в тест

ASP.NET MVC

Jul 20 20:59

ссылка

Комментарий от brave_t:

Опечатка в отличии от

Статистика

Тестов: 153, вопросов: 8596.
Пройдено: 464636 / 2269562.

Тест "Паттерны проектирования" - пройден

пройти еще раз

Правильных ответов: 15 / 20 (75 %)

требуется: 15

Тема	Результат
Классификация паттернов	2 / 4 (50 %)
Порождающие паттерны	2 / 2 (100 %)
Паттерны поведения	3 / 4 (75 %)
Структурные паттерны	4 / 4 (100 %)
Остальные	4 / 6 (66 %)

Дата прохождения теста: 20.07.2020 22:42

Отметьте верные утверждения

<input checked="" type="checkbox"/> <input checked="" type="checkbox"/> Абстрактная фабрика основана на композиции, ее задача - создание семейств взаимосвязанных объектов.	1681 / 2537
<input checked="" type="checkbox"/> <input checked="" type="checkbox"/> Фабричный метод основан на наследовании, его задача - перемещение создания экземпляров в subclasses.	1671 / 2537
<input type="checkbox"/> Фабричный метод основан на композиции, его задача - создание семейств взаимосвязанных объектов.	632 / 2537
<input type="checkbox"/> Абстрактная фабрика основана на наследовании, ее задача - перемещение создания экземпляров в subclasses.	718 / 2537

Пояснение: Абстрактная фабрика основана на композиции: создание объектов реализуется в методе, доступ к которому осуществляется через интерфейс. Фабричный метод основан на наследовании: создание объектов делегируется subclasses, реализующим фабричный метод для создания объектов.

Комментировать вопрос

 (всего: 3)

Автор: GalinaU

Какие основные задачи решает применение паттернов проектирования?

<input checked="" type="checkbox"/> <input checked="" type="checkbox"/> Показываются отношения и взаимодействия между классами или объектами.	4459 / 6745
<input checked="" type="checkbox"/> <input checked="" type="checkbox"/> Описывается решение целого класса абстрактных проблем.	5389 / 6745
<input checked="" type="checkbox"/> <input checked="" type="checkbox"/> Производится унификация терминологии, названий модулей и элементов проекта.	4166 / 6745
<input checked="" type="checkbox"/> <input checked="" type="checkbox"/> Облегчается дискуссия об абстрактных структурах данных между разработчиками.	4401 / 6745

Пояснение: 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.

Комментировать вопрос

 (всего: 5)

Автор: brat

Какие группы паттернов существуют по версии GoF?

<input checked="" type="checkbox"/> <input checked="" type="checkbox"/> Structural	6100 / 6679
<input type="checkbox"/> Reusable	349 / 6679
<input type="checkbox"/> Core	335 / 6679
<input checked="" type="checkbox"/> <input checked="" type="checkbox"/> Creational	5822 / 6679
<input type="checkbox"/> Organizational	898 / 6679
<input checked="" type="checkbox"/> <input checked="" type="checkbox"/> Behavioral	5933 / 6679

Комментировать вопрос

 (всего: 1)

Автор: c0nst

Когда следует использовать паттерн "приспособленец"?

Классификация паттернов

<input checked="" type="checkbox"/> <input checked="" type="checkbox"/> когда большинство состояний объектов могут быть сохранены на диске или рассчитаны во время исполнения	715 / 2169
<input checked="" type="checkbox"/> <input checked="" type="checkbox"/> когда нужно сократить затраты при работе с большим количеством мелких объектов.	1264 / 2169
<input type="checkbox"/> когда объект может иметь несколько представлений	739 / 2169
<input type="checkbox"/> когда нужно изменить реализацию без изменения абстракции	618 / 2169

Комментировать вопрос

 (всего: 5)

Автор: levilaulada

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

Порождающие паттерны

<input type="radio"/> Bridge	166 / 6925
<input checked="" type="radio"/> <input checked="" type="radio"/> Singleton	6368 / 6925
<input type="radio"/> Adapter	140 / 6925
<input type="radio"/> Proxy	157 / 6925
<input type="radio"/> Decorator	74 / 6925

Комментировать вопрос

 (всего: 0)

Автор: c0nst

Какое назначение паттерна Factory Method?

<input checked="" type="radio"/> <input checked="" type="radio"/> Определяет интерфейс для создания объекта, но оставляет подклассам решение о том, какой класс инстанцировать. Этот паттерн позволяет классу делегировать инстанцирование подклассам.	2172 / 3248
<input type="radio"/> Отделяет конструирование сложного объекта от его представления, так что в результате одного и того же процесса конструирования могут получаться разные представления	358 / 3248
<input checked="" type="radio"/> Предоставляет интерфейс для создания семейств взаимосвязанных или взаимозависимых объектов, не специфицируя их конкретные подклассов.	525 / 3248
<input type="radio"/> Задает виды создаваемых объектов с помощью экземпляра-прототипа и создает новые объекты путем копирования этого прототипа.	172 / 3248

Комментировать вопрос

 (всего: 0)

Автор: ruslan5t

Какие из перечисленных паттернов проектирования относятся к порождающим?

Классификация паттернов

<input checked="" type="checkbox"/> <input checked="" type="checkbox"/> Prototype	3307 / 4272
<input checked="" type="checkbox"/> <input checked="" type="checkbox"/> Singleton	3253 / 4272
<input type="checkbox"/> Command	201 / 4272
<input type="checkbox"/> Strategy	277 / 4272
<input checked="" type="checkbox"/> <input checked="" type="checkbox"/> Factory	3814 / 4272

Пояснение: К порождающим шаблонам относятся: Abstract Factory (Абстрактная фабрика), Factory (Фабрика), Builder (Строитель), Singleton (Одиночка), Prototype (Прототип), Factory

Method (Фабричный метод).
Strategy, Command относятся к шаблону поведения.

Комментировать вопрос (всего: 5)

Автор: TestTol

Какой паттерн проектирования используется для создания семейств зависимых между собой объектов?	Порождающие паттерны
<input type="radio"/> Dependency Injection	1043 / 7196
<input type="radio"/> Builder	673 / 7196
<input type="radio"/> Prototype	476 / 7196
<input type="radio"/> Factory Method	688 / 7196
<input checked="" type="radio"/> Abstract Factory	4279 / 7196

Комментировать вопрос (всего: 1)

Автор: admin

Какой из перечисленных паттернов применяется в ситуации, когда нужно произвести некоторые действия над всеми объектами структуры в зависимости от фактического типа каждого объекта?	Паттерны поведения
<input type="radio"/> Iterator	926 / 4337
<input checked="" type="radio"/> Visitor	1801 / 4337
<input type="radio"/> Strategy	707 / 4337
<input type="radio"/> Composite	619 / 4337
<input type="radio"/> State	254 / 4337

Комментировать вопрос (всего: 1)

Автор: alenaiva

Какой шаблон проектирования можно определить следующей фразой: " .. - поведенческий шаблон проектирования, обеспечивающий взаимодействие множества объектов, формируя при этом слабую связанность и избавляя объекты от необходимости явно ссылаться друг на друга "?	
<input checked="" type="radio"/> Mediator	3595 / 7611
<input type="radio"/> Low Coupling	1884 / 7611
<input type="radio"/> High Cohesion	346 / 7611
<input type="radio"/> Information Expert	244 / 7611
<input type="radio"/> Observer	1431 / 7611

Комментировать вопрос (всего: 2)

Автор: gurn

Сколько существует разных типов (и соответственно принципиально разных способов реализации) паттерна Adapter?	Структурные паттерны
<input type="radio"/> 1	767 / 4317
<input checked="" type="radio"/> 2	2687 / 4317
<input type="radio"/> 3	634 / 4317
<input type="radio"/> 4	184 / 4317

Пояснение: Существует ClassAdapter, где используется множественное наследование, и ObjectAdapter, где используется композиция.

Комментировать вопрос (всего: 2)

Автор: c0nst

Какие из перечисленных утверждений верны по отношению к паттерну проектирования Composite:	Структурные паттерны
<input type="checkbox"/> Composite паттерн содержит в себе несколько моделей поведения объекта и предоставляет возможность переключаться между ними.	755 / 4469
<input checked="" type="checkbox"/> Паттерн Composite позволяет работать с композитными и простыми объектами одинаковым образом.	2919 / 4469
<input checked="" type="checkbox"/> Паттерн Composite организует объекты в древовидные структуры.	2862 / 4469
<input type="checkbox"/> Composite паттерн упрощает работу с группой объектов не связанных типов.	1317 / 4469

Комментировать вопрос (всего: 2)

Автор: admin

Целью какого паттерна является расширение функциональности класса или же ее изменение без использования механизма наследования?	Структурные паттерны
<input type="radio"/> Proxy	325 / 4271
<input type="radio"/> Composite	648 / 4271
<input type="radio"/> Flyweight	232 / 4271
<input checked="" type="radio"/> Decorator	3044 / 4271

Пояснение: Структурный паттерн Decorator используется в случаях, когда необходимо без применения механизма наследования расширить функциональность класса или же изменить ее. Другими словами появляется альтернатива наследованию, причем классы не закреплены жестко в иерархии.

Комментировать вопрос (всего: 2)

Автор: nobleclone

Какие из этих паттернов являются паттернами создания объектов?	Классификация паттернов
<input checked="" type="checkbox"/> Abstract Factory	4212 / 4529
<input type="checkbox"/> Adapter	383 / 4529
<input type="checkbox"/> Decorator	582 / 4529
<input type="checkbox"/> Chain of Responsibility	417 / 4529

Комментировать вопрос (всего: 0)

Автор: admin

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.	Паттерны поведения
<input checked="" type="radio"/> да	3284 / 4270
<input type="radio"/> нет	959 / 4270

Пояснение: см. паттерны поведения

Комментировать вопрос (всего: 1)

Автор: Alenka

При малом количестве объектов лучше использовать паттерн Facade вместо Mediator.	Классификация паттернов
<input type="radio"/> Да	1612 / 4536
<input checked="" type="radio"/> Нет	2891 / 4536

Пояснение: Эти паттерны относятся к разным категориям (Facade - структурный, Mediator - поведенческий) и между собой абсолютно не связаны.

Комментировать вопрос (всего: 3)

Автор: admin

Назначением какого паттерна проектирования является предоставление удобного интерфейса к громоздкому и сложному API?	Структурные паттерны
--	----------------------

Добавление вопросов открыто для теста "Паттерны проектирования в Java". Этот тест временно закрыт для добавления, но в скором времени откроем и его.

[ответить](#)

Добавить комментарий

Отправить