Главная

Статьи

Оставить отзыв 🥯

О проекте

Приглашение 🖂

Профиль

### Топ контрибуторов

За последние сутки

Нет данных

Получение Q

Знаете ли Вы, что

Список полученных сертификатов находится на странице Вашего профиля. Сертификаты можно распечатать или разместить на Вашем сайте.

### Опросы

Какая область из перечисленных Вам наиболее интересна?

- С Мобильные технологии
- С Тестирование ПО
- С Защита информации
- С SEO / Интернет маркетинг
- С Компьютерные сети

Ответить

#### Лента обновлений

Jul 23 01:57

ссылка

Комментарий от GeorgeMayoh: Вообще я слышал такую версию: Если класс просто

Jul 22 21:30

соде...

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 отличиЕ от

# Тест "ООП - Основы" - не пройден пройти еще раз

Правильных ответов: 11 / 20 (55 %) требуется: 15

Дата прохождения теста: 21.07.2020 01:08

✓ Как в терминах ООП называется объект, который не может быть изменён после создания?

C Internal object	1159 / 26028
C Abstract object	1602 / 26028
	17003 / 26028
C Sealed object	2120 / 26028
С Все варианты	401 / 26028
С Среди вариантов нет правильного ответа	3651 / 26028

Комментировать вопрос (всего: 3)

Какими бывают отношения между классами (укажите все подходящие варианты)?

16434 / 17569
9423 / 17569
12967 / 17569
3029 / 17569
5163 / 17569

Пояснение: Еще выделяют отношения: композиция и класс-метакласс.

Комментировать вопрос (всего: 7)

🔀 Выберите наиболее точное определение инкапсуляции:

это процесс сокрытия компонентов данных и кода, реализующего функциональность, за
 интерфейсом, не позволяющим пользователю искажать данные.
 6157 / 17568

 ✓ это механизм, который защищает данные и методы, манипулирующие этими данными, от внешнего вмешательства или неправильного использования.
 10548 / 17568

© это принцип ООП, согласно которому каждый объект может использоваться более чем в одной программе.
 350 / 17568

это механизм, позволяющий создавать классы объектов на основе других классов, расширяя и частично изменяя их функциональность и набор атрибутов.
 466 / 17568

Комментировать вопрос (всего: 20)

Автор: VS\_Revan

Автор: akaRomka

Автор: VS\_Revan

✓ В объектно-ориентированных языках программирования полиморфизм обеспечивается с помощью:

0	Передачи аргументов по ссылке	2934 / 17592
0	Ограничения доступа к полям и методам	1199 / 17592
0	Статических полей	238 / 17592
0	Статических методов	680 / 17592
<b>€</b> ✓	<ul> <li>Виртуальных методов</li> </ul>	12246 / 17592
0	Защищённых полей	240 / 17592

#### Статистика

Тестов: 153, вопросов: 8596. Пройдено: 464636 / 2269562. Пояснение: Полиморфизм обеспечивается с помощью виртуальных методов благодаря возможности их переопределения в производных классах.

Комментировать вопрос (всего: 5)

Автор: kukulya

V	✓ События позволяют (продолжите утверждение)			
	6	классу или объекту уведомлять систему о возникновении каких-либо ситуаций	2585 / 12034	
		<ul> <li>классу или объекту уведомлять другие классы или объекты о возникновении ка гуаций</li> </ul>	аких-либо 7722 / 12034	
	0	манипулировать отношением реализации	723 / 12034	
	0	при возникновении каких-либо ситуаций реализовать инкапсуляцию к классу		
	про	ослушивателю	915 / 12034	

Комментировать вопрос (всего: 0)

❤ Чт	о производный класс не наследует от базового класса?	
	<ul> <li>Не наследуются методы конструктора, деструктор, операции присваивания нкции.</li> </ul>	и дружественных 7204 / 13852
0	Не наследуются методы виртуальных функций, статических методов.	3675 / 13852

Комментировать вопрос (всего: 7)

Автор: drabenyuk

✓ High Cohesion (сильное сцепление) - это ОО принцип, наибол	тее
ассоциирующийся с	

Не наследуются методы конструктора и виртуальных методов.

0	сокрытием реализации	2163 / 25490
@ <b>~</b>	тем, что класс спланирован с единственным и конкретным назначением	14616 / 25490
0	тем, как много классы знают про другие только через их API	6145 / 25490
0	разрешением одному объекту быть видимым как разные типы	2362 / 25490

Комментировать вопрос (всего: 13)

Автор: art

Автор: NaGuWaL

2922 / 13852

## Делегат - \_\_\_\_\_ . укажите все верные утверждения

✓ это тип, который определяет сигнатуру метода и может обеспечивать связь с любым методом с совместимой сигнатурой 7372 / 11945
 ✓ используется для передачи методов в качестве аргументов к другим методам. 7877 / 11945
 □ это модификатор уровня агрегируемого метода 1315 / 11945
 □ это не явная реализация полиморфизма 2642 / 11945

Комментировать вопрос (всего: 5) Автор: NaGuWaL

# ✓ Термин «SOLID» представляет собой аббревиатуру пяти важнейших принципов работы с классами в объектно-ориентированном проектировании и включает:

	5044 / 7190
	5810 / 7190
	4712 / 7190
	5475 / 7190
	4639 / 7190
Принцип реализации абстракции	2191 / 7190
Принцип единственного наследования	1724 / 7190

Пояснение: SOLID это принципы: "S"ingle responsibility - единственной обязанности "O"penclosed - открытости/закрытости "L"iskov substitution - подстановки Барбары Лисков "I"nterface

Ко	мментировать вопрос (всего: 2)	Автор: tehnil
<b>~</b>	Виртуальные методы Необходимо продолжить утверждение (укажите все подходящ варианты).	цие
	служат исключительно для реализации полиморфизма	3013 / 17420
	✓ позволяют наследникам переопределять методы родителя	13215 / 17420
	не реализуются в базовом классе	4295 / 17420
	✓ используются при реализации полиморфизма	13507 / 17420
Ко	мментировать вопрос (всего: 13)	Автор: Evguenitch
×	Перечислите недостатки ООП парадигмы.	
		19514 / 2452
	Невозможность повторного использования кода	941 / 24522
	Г ✓ Избыточность	15680 / 24522
	Недостаточная гибкость в создании иерархии	4101 / 24522
	Невозможность абстракции	867 / 24522
	Пояснение: У каждой парадигмы программирования есть свои положительные отрицательные стороны. Отрицательные стороны ОО подхода - это бОльшие требования к ресурсам с же избыточность при описании сущностей.	
	мментировать вопрос (всего: 17)  Класс в ООП является абстрактным типом данных.   Не верно.	Автор: ss18 2975 / 7503
	Класс в ООП является абстрактным типом данных.	
	Класс в ООП является абстрактным типом данных.  • Не верно.	2975 / 750: 4513 / 750:
**	Класс в ООП является абстрактным типом данных.  Не верно.  Верно.  Пояснение: Класс — разновидность абстрактного типа данных, характеризуе	2975 / 750: 4513 / 750:
<b>36</b>	Класс в ООП является абстрактным типом данных.  Не верно.  Пояснение: Класс — разновидность абстрактного типа данных, характеризуе своего построения.  мментировать вопрос (всего: 7)  Выберите правильные утверждения по отношению к параметри полиморфизму:  Поддерживается во многих языках посредством перегрузки функций и мето.  ✓ Позволяет определять функцию или тип данных обобщённо, так что значен	2975 / 750: 4513 / 750: емый способом Автор: tehnil IЧЕСКОМУ дов 4442 / 10574
<b>36</b>	Класс в ООП является абстрактным типом данных.  Не верно.  Пояснение: Класс — разновидность абстрактного типа данных, характеризуе своего построения.  мментировать вопрос (всего: 7)  Выберите правильные утверждения по отношению к параметри полиморфизму:  Поддерживается во многих языках посредством перегрузки функций и мето Упозволяет определять функцию или тип данных обобщённо, так что значен обрабатываются идентично вне зависимости от их типа	2975 / 7503 4513 / 7503 емый способом Автор: tehnil 1ческому дов 4442 / 10574
<b>36</b>	Класс в ООП является абстрактным типом данных.  В Не верно.  Пояснение: Класс — разновидность абстрактного типа данных, характеризуе своего построения.  Мментировать вопрос (всего: 7)  Выберите правильные утверждения по отношению к параметри полиморфизму:  Поддерживается во многих языках посредством перегрузки функций и мето.  № ✓ Позволяет определять функцию или тип данных обобщённо, так что значен обрабатываются идентично вне зависимости от их типа.  Позволяет давать одинаковые имена программным сущностям с различны.  ✓ Доступен в нескольких объектно-ориентированных языках, где он часто иде	2975 / 7503 4513 / 7503 емый способом Автор: tehnil 1Ческому дов 4442 / 10574 ния 5428 / 10574 м по <b>веде</b> ни <b>ю</b> 574
<b>36</b>	Класс в ООП является абстрактным типом данных.  Не верно.  Пояснение: Класс — разновидность абстрактного типа данных, характеризуе своего построения.  Мментировать вопрос (всего: 7)  Выберите правильные утверждения по отношению к параметри полиморфизму:  Поддерживается во многих языках посредством перегрузки функций и мето У Позволяет определять функцию или тип данных обобщённо, так что значен обрабатываются идентично вне зависимости от их типа  Позволяет давать одинаковые имена программным сущностям с различны	2975 / 7503 4513 / 7503 емый способом Автор: tehnil 1Ческому дов 4442 / 10574 ния 5428 / 10574
₩ Kor	Класс в ООП является абстрактным типом данных.  Не верно.  Пояснение: Класс — разновидность абстрактного типа данных, характеризуе своего построения.  Мментировать вопрос (всего: 7)  Выберите правильные утверждения по отношению к параметри полиморфизму:  Поддерживается во многих языках посредством перегрузки функций и мето обрабатываются идентично вне зависимости от их типа  Позволяет давать одинаковые имена программным сущностям с различны обрабатываются идентично вне зависимости от их типа  Озволяет давать одинаковые имена программным сущностям с различны обрабатываются идентично вне зависимости от их типа обрабатываются и и и и и и и и и и и и и и и и и и и	2975 / 750: 4513 / 750: 2966
<b>¥</b> Kor <b>¥</b>	Класс в ООП является абстрактным типом данных.	2975 / 7503 4513 / 7503 емый способом Автор: tehnil  149СКОМУ  дов 4442 / 10574  ния 5428 / 10574 ет под названием 5243 / 10574  Автор: nvgup
₩ Kor	Класс в ООП является абстрактным типом данных.	2975 / 7503 4513 / 7503 емый способом Автор: tehnil  149СКОМУ  дов 4442 / 10574  ния 5428 / 10574 ет под названием 5243 / 10574  Автор: nvgup
<b>¥</b> Kor <b>¥</b>	Класс в ООП является абстрактным типом данных.  Не верно.  Пояснение: Класс — разновидность абстрактного типа данных, характеризуе своего построения.  Мментировать вопрос (всего: 7)  Выберите правильные утверждения по отношению к параметри полиморфизму:  Поддерживается во многих языках посредством перегрузки функций и мето У Позволяет определять функцию или тип данных обобщённо, так что значен обрабатываются идентично вне зависимости от их типа  Позволяет давать одинаковые имена программным сущностям с различны У Доступен в нескольких объектно-ориентированных языках, где он часто иде "дженерик" или "шаблоны"  Не поддерживается в языках Java и C++  мментировать вопрос (всего: 3)  Наследует ли производный класс перегруженные операторы бакласса?	2975 / 7503 4513 / 7503 4513 / 7503  мый способом  Автор: tehnil  1ческому  дов 4442 / 10574  ния 5428 / 10574  ет под названием 5243 / 10574  Автор: nvgup

создании иерархии объектов используется либо наследование, либо

<ul> <li>Какое наиболее важное преимущество предоставляет персистентност</li> </ul>	Ь
объектов в приложении?	

Персистентность делает систему более защищенной от внешних атак
 2437 / 25632

Персистентность делает исходный код системы простым в понимании и обеспечивает
 повторное использование кода
 4446 / 25632

Персистентность позволяет системе быстрее стартовать
 1270 / 25632

№ Персистентность позволяет хранить и восстанавливать состояние объектов между
 выключениями системы
 14688 / 25632

Персистентность увеличивает скорость поиска объектов по атрибутам 2218 / 25632

Комментировать вопрос (всего: 4)

Как называется объект, задача которого состоит в реакции на возникновение определенного события?

	0	Repeater	327 / 19912
	0	Callback	3651 / 19912
	@ <b>~</b>	Observer	8852 / 19912
	0	Publisher	526 / 19912
	0	Trigger	6491 / 19912
Ко	ммен	тировать вопрос (всего: 18)	Автор: admin

🧗 Выберете наиболее точное определение полиморфизму:

это механизм, позволяющий создавать классы объектов на основе других классов, расширяя и частично изменяя их функциональность и набор атрибутов. 2825 / 17407

⊙ это принцип ООП, согласно которому каждый объект может использоваться более чем в одной программе.
 729 / 17407

С ✓ это принцип, согласно которому объекты, имеющие одинаковый интерфейс, могут вести себя по-разному
12803 / 17407

это процесс сокрытия компонентов данных и кода, реализующего функциональность, за интерфейсом, не позволяющим пользователю искажать данные.

450 / 17407

© это механизм, который объединяет данные и методы, манипулирующие этими данными, и защищает и то и другое от внешнего вмешательства или неправильного использовани №85 / 17407

Комментировать вопрос (всего: 6)

Автор: VS\_Revan

Автор: admin

```
🛩 Какой принцип ООП "нарушает" следующий фрагмент кода:
```

```
class Counter {
    public int count;
    public void increment() { count++; }
    public int get() { return count; }
}

public class Main {
    public static void main(String[] args) {
        Counter counter = new Counter();
        counter.count = 5;
    }
}
```

С Композиция 1412 / 26168

Полиморфизм 1395 / 26168

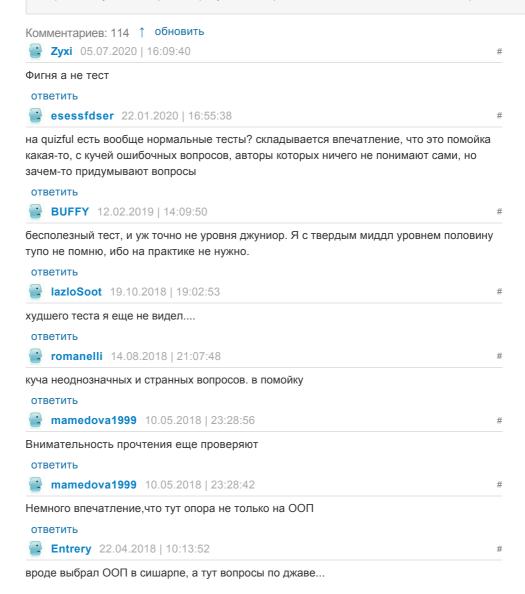
мменти	ировать вопрос (всего: 8)	Автор: kukulya
	вение: Модификаторы доступа запрещают (или ограни оянию класса (и его экземпляров).	чивают) доступ к внутреннему
<u> </u>	Динамическое выделение памяти	1594 / 2536
	Обработка исключений	1271 / 2536
	Статические методы	4255 / 2536
	Виртуальные методы	4954 / 2536
<b>⋈ ✓</b> [	Модификаторы доступа	23002 / 2536
	е механизмы в ОО языках обычно позволяют ектов?	обеспечить инкапсуляцию
мменти	ировать вопрос (всего: 6)	Автор: admir
	Агрегация	1312 / 26168
@ < 1	<i>И</i> нкапсуляция	21282 / 26168
<b>6</b> F	Равенство	639 / 26168

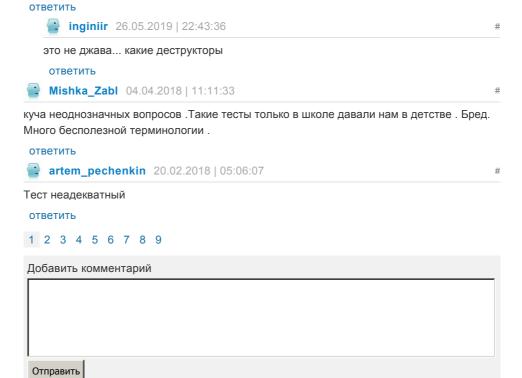
#### Предложить свой вопрос

#### ShareTweet

**Важно:** Ниже обсуждается только тест в целом: покрытие тематик, типы вопросов, пути улучшения теста и так далее.

Замечания, касающиеся отдельных вопросов теста, просьба оставлять в соответствующих топиках, которые доступны на странице результатов прохождения теста возле каждого вопроса.







Наши представительства





