



Онлайн образование

otus.ru



Проверить, идет ли запись

Меня хорошо видно && слышно?



Правила вебинара



Активно
участвуем



Off-topic обсуждаем
в учебной группе в
Telegram



Задаем
вопрос в чат



Вопросы вижу в чате,
могу ответить не сразу

Тема вебинара

Разработчик C++ - базовый курс

Модуль 8: Разное и дополнительное

Идиомы и полезные практики C++

Часть 2

Пальчуковский Евгений



Разработчик ПО

Развиваю технологии финансовых услуг с помощью C++

Email: eugene@palchukovsky.com

Telegram: @palchukovsky

Маршрут вебинара

1. Self-assignment check

2. PImpl

3. Virtual constructor

4. Mixin

5. “Мюнхаузен” (enable_shared_from_this)



Цели вебинара

После занятия вы сможете

1. Оpoznать в коде популярные идиомы
2. Рассказать, для чего они нужны
3. Правильно применять их на практике

Смысл

Зачем вам это уметь

1. Читать идиоматичный код
 2. Писать идиоматичный код
-

Self-assignment check

- Позволяет избежать проблем при присваивании **самому себе**
- Хорошо сочетается с copy and swap



Pointer to implementation

Pimpl /пимпл/

- Скрывает реализацию класса
- Позволяет управлять включениями (include)
- Позволяет ускорять время сборки большого проекта
- Позволяет поставлять стабильный бинарный интерфейс
- Бонус: облегчает реализацию copy and swap



Virtual constructor

- А так можно в C++?
- Позволяет делать полиморфное создание объектов



Mixin

- Наследует поведение
- Позволяет управлять поведением на этапе компиляции
- ... при этом нет накладных расходов, связанных с виртуальными вызовами



enable_shared_from_this

"Мюнхаузен"



8

Объект может сам себе продлить время жизни



Заключение

Цели вебинара

После занятия вы сможете

1. Оpoznать в коде популярные идиомы
2. Рассказать, для чего они нужны
3. Правильно применять их на практике

Смысл

Зачем вам это уметь

1. Читать идиоматичный код
 2. Писать идиоматичный код
-

Список материалов для изучения

1. **Скотт Маейрс**. Эффективное использование C++
2. **Скотт Маейрс**. Наиболее эффективное использование C++
3. **Скотт Маейрс**. Современный и эффективный C++
4. **Fedor Pikus**. Hands-On Design Patterns with C++
5. **Jason Turner**. C++ Best Practices

Вопросы?



Ставим “+”,
если вопросы есть



Ставим “-”,
если вопросов нет

**Заполните, пожалуйста,
опрос о занятии
по ссылке в чате**