

Принципы проектирования

Принципы проектирования DRY, KISS, SOLID, GRASP.

План урока

1. Принципы проектирования.
2. BASE, SOLID, GRASP.

Принципы проектирования

Проектирование

Создание спецификации продукта, предназначенного для достижения заявленных целей, используя набор примитивных компонентов, имеющих заданные ограничения.



Общие принципы проектирования

- Рассмотреть альтернативные подходы
- Декомпозиция
- Повторное использование блоков
- Сокращать «интеллектуальную дистанцию»
- Единообразие и целостность
- Возможность внесения изменений
- Проектирование ≠ кодирование

Методы проектирования

1. Абстракция.
2. Декомпозиция.
3. Модульность.
4. Модель.
5. Тестирование.

Принцип DRY

Don't Repeat Yourself — «Не повторяйся»

«Каждая часть знания должна иметь единственное, непротиворечивое и авторитетное представление в рамках системы».

Э. Хант и Д. Томас

Принцип KISS

**Keep It Simple Stupid — «Будь
проще»**

«Не следует множить сущее без
необходимости».

«Бритва Оккама»



Принципы SOLID

Роберт Мартин ака «Uncle Bob»



Принципы SOLID

1. Принцип единственной ответственности (The **S**ingle Responsibility Principle).
2. Принцип открытости / закрытости (The **O**pen Closed Principle).
3. Принцип подстановки Барбары Лисков (The **L**iskov Substitution Principle).
4. Принцип разделения интерфейса (The **I**nterface Segregation Principle).
5. Принцип инверсии зависимостей (The **D**ependency Inversion Principle).

Принципы GRASP

GRASP = General Responsibility
Assignment Software Patterns
(Общие шаблоны распределения
ответственностей в ПО).

Принципы GRASP

1. Информационный эксперт (Information expert).
2. Создатель (Creator).
3. Контроллер (Controller).
4. Слабая степень связности (Low Coupling).
5. Высокое сцепление (High cohesion).
6. Полиморфизм (Polymorphism).
7. Чистая выдумка (Pure Fabrication).
8. Посредник (Indirection).
9. Устойчивость к изменениям (Protected Variations).

Базовые и включенные шаблоны

```
1 from jinja2 import FileSystemLoader
2 from jinja2.environment import Environment
3
4
5 def render(template_name, folder='templates', **kwargs):
6     env = Environment()
7     # указываем папку для поиска шаблонов
8     env.loader = FileSystemLoader(folder)
9     # находим шаблон в окружении
10    template = env.get_template(template_name)
11    return template.render(**kwargs)
12
```

Практическое задание

**В этой
самостоятельной
работе
тренируем
умения:**

Работать с шаблонизатором

**Использовать базовые и
включённые шаблоны**

Зачем:

Чтобы избежать дублирования в шаблонах

Последовательность действий:

- Внести в WSGI-фреймворк изменения, которые позволят использовать механизм наследования и включения шаблонов.
- Создать базовый шаблон для всех страниц сайта.
- Если нужно, создать один или несколько включённых шаблонов
- Добавить на сайт меню, которое будет отображаться на всех страницах.
- Улучшить существующие страницы с использованием базовых и включённых шаблонов.
- Проверить, что фреймворк готов для дальнейшего использования (при желании добавить какой-либо полезный функционал).

Спасибо!
Каждый день
вы становитесь
лучше :)

