

Объектно-ориентированное программирование

Классы, атрибуты, методы,
конструкторы, деструкторы



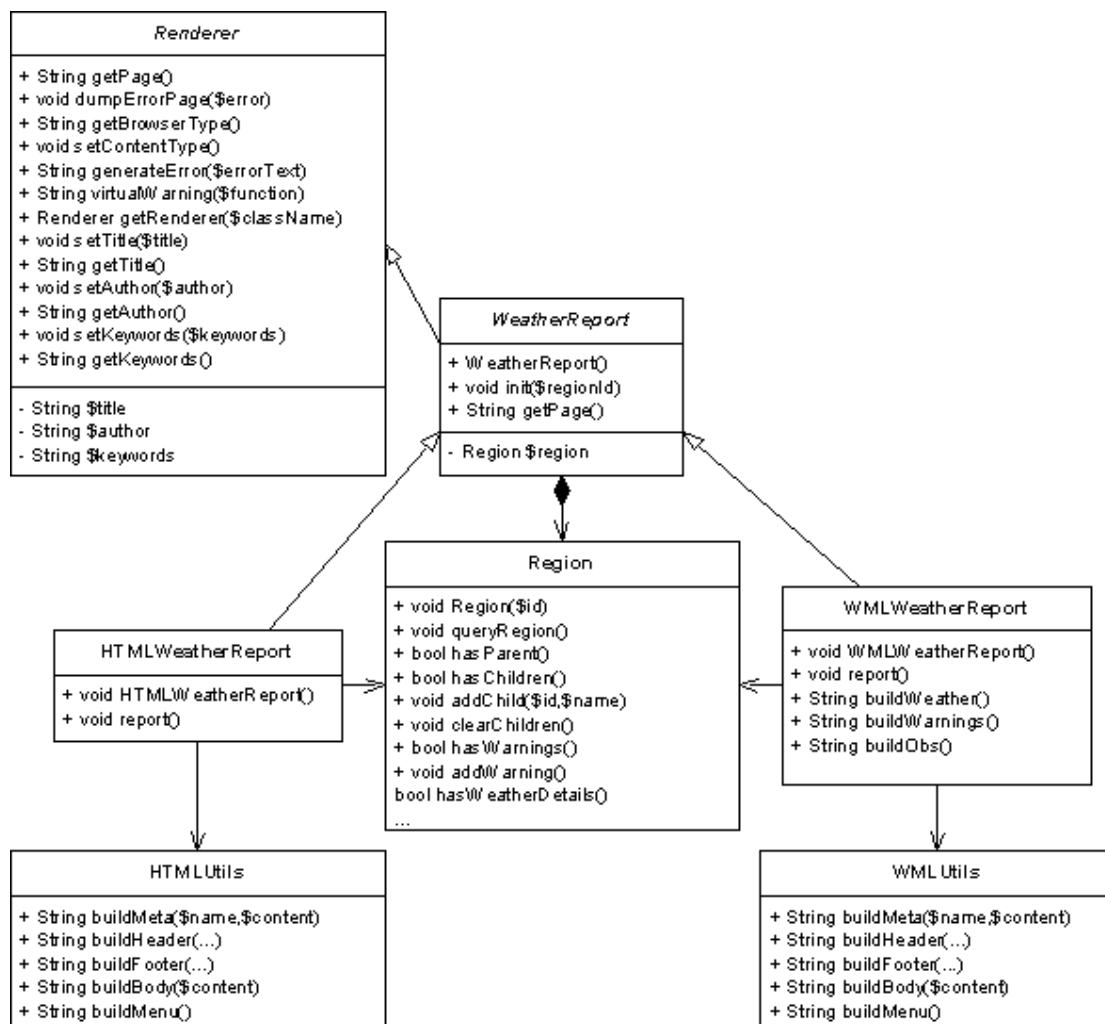
Что такое ООП.

Объектно-ориентированное программирование - методология программирования при котором программа представляет собой совокупность взаимодействующих объектов, каждый из которых является экземпляром определенного класса.

Есть 4 основных принципа ООП:

- Инкапсуляция
- Наследование
- Полиморфизм
- Абстракция

Последний принцип мы не будем рассматривать. Он мутный и настолько фундаментален что будем им пользоваться не проговаривая, что мы это делаем





Классы и их атрибуты

Классы - модель для создания объектов определенного типа. Также классы, кто-то называет "пользовательскими типами". По своей сути похож на структуры. Но есть некоторая разница.

Объект - используемый экземпляр какого-либо класса. В процессе обучения будем называть. Можно сказать что объект "А" принадлежит классу "В". К колайдеру коду!

Атрибут/поле класса - данные класса, описание которых задает программист когда описывает класс. К коду!



Модификаторы доступа. Инкапсуляция.

Основная разница между структурой и классом в том, что к полям структуры по умолчанию можно обратиться из любой части программы, в которой можно создать объект структуры. К коду!

Существует способ “открыть” атрибуты для всех пользователей класса. Ключевое слово **public** O
таких атрибутах говорят что они “публичные” или “открытые”.

Также существует модификатор с обратным эффектом: **private** O
таких атрибутах говорят, что они “приватные” или “открытые”.

По умолчанию к полям класса нельзя обратиться, а также считается, что все атрибуты и должны быть приватными. В чем тогда смысл таких атрибутов? Смысл в [инкапсуляции](#)



Инкапсуляция. Инкапсуляция. Инкапсуляция

Инкапсуляция - это сокрытие данных объекта от пользователя класса. Вы будете часто слышать что-то подобное от недоучившихся программистов. Отчасти это правда, но полную суть не отражает. Все равно, что сказать, что люди кушают ради запаха еды.

Чтобы понять что такое инкапсуляция введем понятие метода. Потому что инкапсуляцию обеспечивает связка модификаторов доступа и методов. Метод класса - функция привязанная к объекту класса. Метод класса может обращаться к любому атрибуту этого класса, несмотря на модификаторы доступа. Описание метода как и атрибута тоже задается при описании класса. К коду!



Итак, инкапсуляция

Инкапсуляция - связка методов класса с данными класса. Инкапсуляция позволяет пользователю класса не думать о конкретной внутренней реализации объекта, пользуясь только интерфейсом класса этого объекта (предоставленными для пользователя методами).



Наследование

~~Преследование~~ Наследование - возможность создать новый класс, на основе уже описанного. Наследование позволяет не повторять код, а также обеспечивает следующий принцип ООП - полиморфизм. К коду!



Полиморфизм

Широко распространено определение полиморфизма приписываемое создателю языка C++, Страуструпу - “Один интерфейс - множество определений”. Что это значит?

К коду!



Эпитафия

ООП - это навык, которому нужно учиться на практике и не меньше знаниями, о которых нужно читать как теорию. Если обо всем, что мы изучали до этого, можно было сказать что оптимальным вариантом является 20% теории и 80% практики, то в случае с объектно-ориентированной парадигмой оптимальным является 50% и того и другого. И сперва теория - потом практика.