1. Придумать класс, который описывает любую сущность из предметной области интернет-магазинов: продукт, ценник, посылка и т.п.

2. Описать свойства класса из п.1 (состояние).

3. Описать поведение класса из п.1 (методы).

4. Придумать наследников класса из п.1. Чем они будут отличаться?

Ничего абстрактного придумать не получается, поэтому скидываю реальную задачу, которую делал на проекте Кволити Моторс. Это отправщик писем должностным лицам компании из форм на сайте, все учетные данные и e-mail адреса изменил, чтобы никого не смущать. Все 4 пункта в рамках той задачи выполнены. Файл архива Ajax.rar. Если будут сомнения в моем авторстве, могу объяснить каждый момент этого программного кода, кроме того, что находится в папке Mailer – это сторонний продукт. Спасибо!

5. Дан код:

class A {

public function foo() {

static $x = 0;

echo ++$x;

}

}

$a1 = new A();

$a2 = new A();

$a1->foo();

$a2->foo();

$a1->foo();

$a2->foo();

Что он выведет на каждом шаге? Почему?

Выведет: 1 2 3 4. Здесь создается статическая переменная в методе foo(), которая принадлежит этому методу. Инициализация такой переменной происходит при первом вызове функции/метода, после чего она сохраняет свое состояние и те изменения, которые производятся с ней в контексте исполнения её функции/метода - хозяина. В данном случае при выводе значения в поток, мы используем преинкремент, который увеличивает значение переменной на единицу, после чего возвращает её значение.

Немного изменим п.5:

class A {

public function foo() {

static $x = 0;

echo ++$x;

}

}

class B extends A {

}

$a1 = new A();

$b1 = new B();

$a1->foo();

$b1->foo();

$a1->foo();

$b1->foo();

6. Объясните результаты в этом случае.

Выведет 1 1 2 2. Класс В наследует класс А, соответственно и все его методы, объекты $a1 и $b1 создаются на основе этих классов, в результате чего мы получаем два разных экземпляра объектов от двух разных классов, со своими собственными методами foo(), так как метод не является статическим, а значит принадлежи уже объекту. А далее происходит механизм как в предыдущей задаче, только в контексте двух разных методов разных объектов.

7. \*Дан код:

class A {

public function foo() {

static $x = 0;

echo ++$x;

}

}

class B extends A {

}

$a1 = new A;

$b1 = new B;

$a1->foo();

$b1->foo();

$a1->foo();

$b1->foo();

Что он выведет на каждом шаге? Почему?

То же, что и в предыдущей задаче, по тем же самым принципам, т.к. эти два кода практически не отличаются, за исключением того, что в данном случае при создании экземпляров объектов не происходит вызова метода - конструктора.