

1. Сделайте объект какого-нибудь класса. Примените к объекту функцию `get_class` и узнайте имя класса, которому принадлежит объект.
2. Сделайте два класса: `Test1` и `Test2`. Пусть оба класса имеют свойство `name`.
3. Создайте некоторое количество объектов этих классов и запишите в массив `$arr` в произвольном порядке. Переберите этот массив циклом и для каждого объекта выведите значение его свойства `name` и имя класса, которому принадлежит объект.
4. `method1`, `method2` и `method3`. С помощью функции `get_class_methods` получите массив названий методов класса `Test`.
5. Создайте объект класса `Test`, запишите его в переменную `$test`. С помощью функции `get_class_methods` получите массив названий методов объекта. Переберите его циклом и в цикле вызовите каждый метод класса, используя объект `$test`. Переберите этот массив циклом и в этом цикле вызовите каждый метод объекта.
6. Сделайте класс `Test` с публичными свойствами `prop1` и `prop2`, а также с приватными свойствами `prop3` и `prop4`.
7. класса `Test`. Выведите массив доступных свойств.
8. Вызовите функцию `get_class_vars` внутри класса `Test` (например, в конструкторе). Выведите массив доступных свойств.
9. Сделайте класс `Test` с публичными свойствами `prop1` и `prop2`, а также с приватными свойствами `prop3` и `prop4`. Создайте объект этого класса. С помощью функции `get_object_vars` получите массив свойств созданного объекта.
10. Пусть у вас есть класс `Test1` и нет класса `Test2`. Проверьте, что выведет функция `class_exists` для класса `Test1` и для класса `Test2`.
11. Пусть GET параметром в адресную строку передается название класса. Проверьте, существует ли такой класс. Выведите соответствующее сообщение на экран.
12. Сделайте класс `Test` с методом `method1` и без метода `method2`. Проверьте, что выведет функция `method_exists` для метода `method1` и для метода `method2`.
13. Пусть GET параметрами в адресную строку передаются название класса и его метод. Проверьте, существует ли такой класс. Если существует - проверьте существование переданного метода. Если и метод существует - создайте объект данного класса, вызовите указанный метод и выведите результат его работы на экран.
14. Сделайте класс `Test` со свойством `prop1` и без свойства `prop2`. Проверьте, что выведет функция `property_exists` для свойства `prop1` и для свойства `prop2`.
15. Дан массив со свойствами класса. Дан также класс, имеющий часть из этих свойств. Переберите этот массив циклом, для каждого свойства проверьте, существует ли оно в классе и, если существует, выведите на экран значение этого свойства.
16. Сделайте класс `ChildClass` наследующий от `ParentClass`. С помощью функции `get_parent_class` выведите на экран родителя класса `ParentClass`.
17. Сделайте класс `ChildClass` наследующий от `ParentClass`, который в свою очередь наследует от `GrandParentClass`.
18. С помощью функции `is_subclass_of` проверьте, является ли класс `ChildClass` потомком `GrandParentClass`.
19. С помощью функции `is_subclass_of` проверьте, является ли класс `ParentClass` потомком `GrandParentClass`.
20. С помощью функции `is_subclass_of` проверьте, является ли класс `ChildClass` потомком `ParentClass`.
21. Сделайте класс `ChildClass` наследующий от `ParentClass`. Создайте объект класса `ChildClass`, запишите его в переменную `$obj`.
22. С помощью функции `is_a` проверьте, принадлежит ли объект `$obj` классу `ChildClass`.
23. С помощью функции `is_a` проверьте, принадлежит ли объект `$obj` классу `ParentClass`.
24. Выведите на экран список всех объявленных классов.