



PHP Level 2

Урок 2

ООП в PHP.

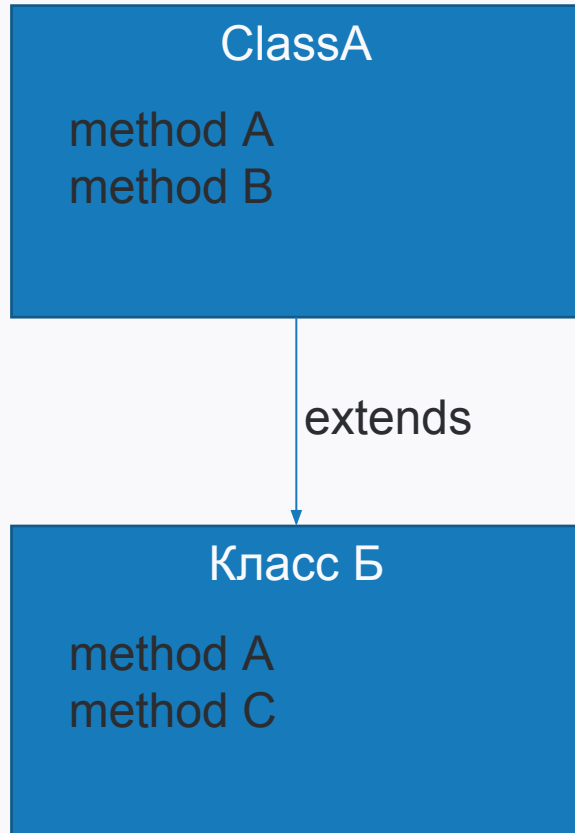
Расширенное изучение.

# План урока

1. Ключевые слова
2. Абстрактные классы и интерфейсы
3. Перегрузка и магические методы
4. Контроль типов и пространства имён
5. Трейты
6. Паттерн Singleton



# Ключевое слово parent



```
<?php
class BaseClass {
    function __construct() {
        echo "Конструктор класса BaseClass\n";
    }
}

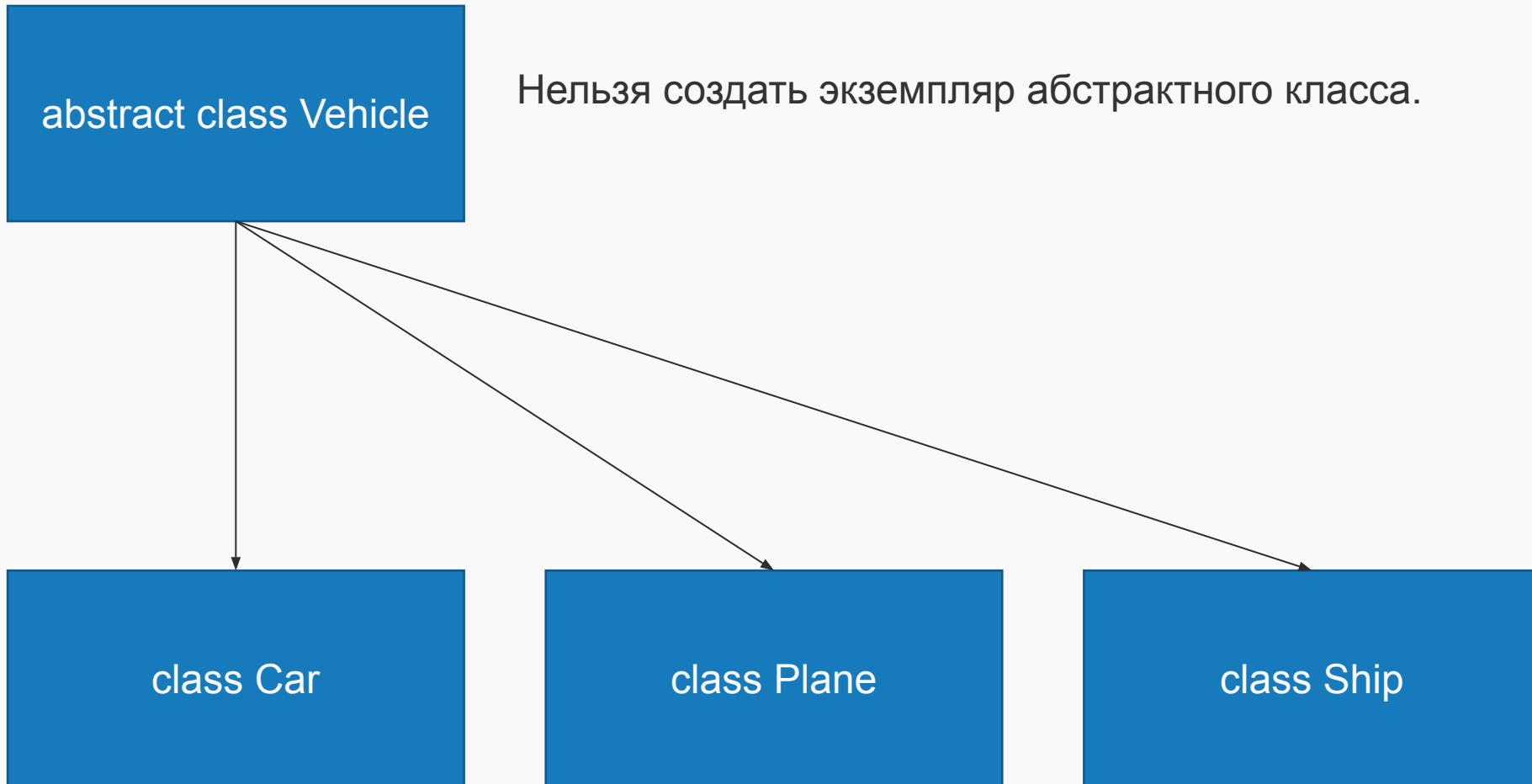
class SubClass extends BaseClass {
    function __construct() {
        parent::__construct();
        echo "Конструктор класса SubClass\n";
    }
}

$obj = new BaseClass();
$obj = new SubClass();

?>
```



# Абстрактные классы



# Интерфейсы

interface CarTemplate

С помощью интерфейсов можно **описать** методы, которые должны быть реализованы в классе без необходимости описания их функционала.

class Audi



# Перегрузка и магические методы

Перегрузка в PHP дает возможность динамически "создавать" свойства и методы. Такие методы и свойства обрабатываются с помощью "волшебных" методов, которые можно создать в классе для различных видов действий.

```
<?php  
  
public void __set (string $name , mixed $value)  
  
public mixed __get (string $name)  
  
?>
```



# Контроль типа

Тип	Описание	Минимальная версия PHP
Имя класса/интерфейса	Аргумент должен быть <a href="#">instanceof</a> , что и имя класса или интерфейса.	PHP 5.0.0
<i>self</i>	Этот параметр должен быть <a href="#">instanceof</a> того же класса, в методе которого он указан. <i>self</i> можно использовать только в функциях класса либо объекта класса.	PHP 5.0.0
<a href="#">array</a>	Аргумент должен быть типа <a href="#">array</a> .	PHP 5.1.0
<a href="#">callable</a>	Аргумент должен быть корректным <a href="#">callable</a> типом.	PHP 5.4.0
<a href="#">bool</a>	Аргумент должен быть типа <a href="#">boolean</a> .	PHP 7.0.0
<a href="#">float</a>	Аргумент должен быть <a href="#">float</a> типа.	PHP 7.0.0
<a href="#">int</a>	Аргумент должен быть типа <a href="#">integer</a> .	PHP 7.0.0
<a href="#">string</a>	Аргумент должен иметь тип <a href="#">string</a> .	PHP 7.0.0



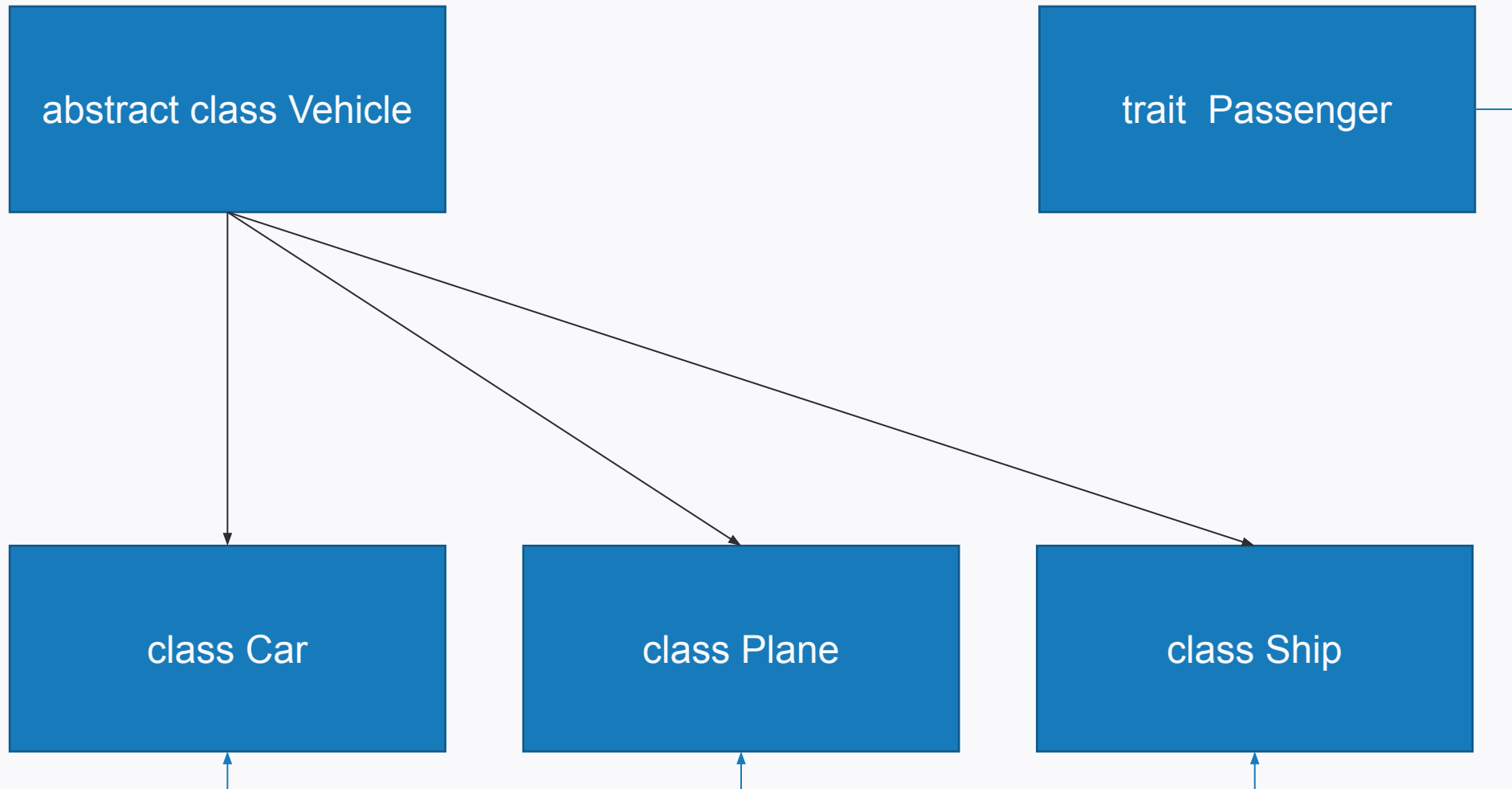
# Пространства имён

Пространства имен, это один из способов инкапсуляции элементов.





# Трейты



# Singleton

**Шаблон проектирования** или **паттерн** в разработке программного обеспечения — повторяемая архитектурная конструкция, представляющая собой решение проблемы проектирования в рамках некоторого часто возникающего контекста.

**Одиночка** (англ. **Singleton**) — шаблон проектирования, гарантирующий, что в однопоточном приложении будет единственный экземпляр класса с глобальной точкой доступа.

