

Интерфейс

В PHP ООП интерфейс - это класс, в котором все методы являются абстрактными и открытыми.

Класс, абстрактный и открытый. Класс представляет собой шаблон или образец объекта, он является основой объектно-ориентированного программирования. Абстрактный - это специальный метод, который объявляется, но не имеет тела, только пустые скобки. Открытый - это модификатор доступа `public`, он связан с доступом к методам. Это означает, что он доступен из любого места: из этого класса, вне его пределов и из любого производного класса.

Интерфейс PHP представляет собой инструмент для создания надежного кода, определяющего методы, которые должны быть реализованы с помощью класса без определения обработки этих методов. Это мощная и простая концепция, используемая в объектно-ориентированном PHP. Для его создания используется ключевое слово `interface`, он определяется так же, как класс PHP. Методы, содержащиеся в нем, не имеют никакого функционала, но интерфейс задает, какие методы необходимы для других классов. Любой класс, наследующийся от интерфейса, должен содержать те же методы, которые содержатся в интерфейсе, иначе выдается ошибка.

В интерфейсе ООП PHP мы задаем только имена методов и их параметры, а реализованы они могут быть позже. Принято объявлять все методы интерфейса как `public`. Для реализации интерфейса используется ключевое слово `implements`. При необходимости класс может реализовать более одного интерфейса, между собой они разделяются запятой. Как и класс, интерфейс может содержать константы. Единственное отличие заключается в том, что они не могут быть переопределены в производном классе.

Интерфейс определяется с помощью ключевого слова `interface`. Все методы в нем должны быть открытыми и абстрактными:

```
interface animal {  
    public function motion();  
}
```

Этот интерфейс может быть реализован любым классом `animal`, все классы `animal` будут содержать метод `motion`:

```
class Lion implements animal {  
    public function motion() {  
        // body  
    }  
}
```

```
class Fish implements animal {  
    public function motion() {  
        // body  
    }  
}
```

PHP-интерфейс и абстрактные классы

Интерфейсы и абстрактные классы очень похожи по своей концепции. Они помогают в создании шаблонов для классов, которые будут использоваться в приложении. Согласно основам ООП PHP, они отличаются друг от друга следующим:

- В интерфейсе все методы являются абстрактными (без реализации). В абстрактном классе лишь некоторые методы являются абстрактными. Абстрактный класс должен содержать, по крайней мере, один абстрактный метод. Иначе это будет стандартный класс PHP;
- В интерфейсе PHP все объявленные методы являются открытыми, а в абстрактном классе методы могут быть открытыми, частными или защищенными. Следовательно, для интерфейсов существует ограничение на использование модификаторов доступа, а в абстрактном классе таких ограничений нет;
- Класс может реализовать неограниченное количество интерфейсов. В то же время класс PHP может породить только один абстрактный класс;
- Вы должны переопределять все методы интерфейса, а в абстрактном классе у вас есть выбор: переопределять методы или оставить их без изменений;
- Интерфейсы имеют более строгие правила, чем абстрактные классы. Интерфейс PHP призван обеспечить определенную логику, он выступает в качестве пустой оболочки или шаблона для других классов.