

OPEN-CLOSED




```
class OrderRepository
{
    public function load($orderId)
    {
        $pdo = new PDO($this->config->getDsn(), $this->config->getDBUser(), $this->config->getDBPassword());
        $statement = $pdo->prepare('SELECT * FROM `orders` WHERE id=:id');
        $statement->execute(array(':id' => $orderId));
        return $query->fetchObject('Order');
    }
    public function save($order){/*...*/}
    public function update($order){/*...*/}
    public function delete($order){/*...*/}
}
```

```
interface IOrderSource
{
    public function load($orderId);
    public function save($order);
    public function update($order);
    public function delete($order);
}
```

```
class MySQLOrderSource implements IOrderSource
{
    public function load($orderId);
    public function save($order) { /* ... */ }
    public function update($order) { /* ... */ }
    public function delete($order) { /* ... */ }
}
```

```
class ApiOrderSource implements IOrderSource
{
    public function load($orderId);
    public function save($order) { /* ... */ }
    public function update($order) { /* ... */ }
    public function delete($order) { /* ... */ }
}
```

```
class OrderRepository
{
    private $source;

    public function setSource(IOrderSource $source)
    {
        $this->source = $source;
    }

    public function load($orderId)
    {
        return $this->source->load($orderId);
    }

    public function save($order){/*...*/}
    public function update($order){/*...*/}
}
```

OPEN - CLOSED

Данный принцип гласит – "программные сущности должны быть открыты для расширения, но закрыты для модификации". На более простых словах это можно описать так – все классы, функции и т.д. должны проектироваться так, чтобы для изменения их поведения, нам не нужно было изменять их исходный код.

```
class OrderRepository
{
    private $source;

    public function setSource(IOrderSource $source)
    {
        $this->source = $source;
    }

    public function load($orderId)
    {
        return $this->source->load($orderId);
    }

    public function save($order){/*...*/}
    public function update($order){/*...*/}
}
```

LSKOV SUBSTITUTION

LSKOV SUBSTITUTION

