

## ИМПЛЕМЕНТАЦИЯ В КЛАССАХ

Простым языком, **имплементация** - это описание того, что должно быть внутри класса при помощи одного или нескольких интерфейсов. В свою очередь, это позволяет вам строить определенные **связи и зависимости**. Реализуется это через ключевое слово **implements**:

```
1 interface IUser {  
2     login: string;  
3     password: string;  
4 }
```

```
1 class UserForm implements IUser {  
2     login: string;  
3     password: string;  
4 }
```

Теперь в классе должны быть все свойства и методы, которые были в интерфейсе. Сам класс **можно** дополнять другими методами или свойствами.

При этом мы создали **связь** и если в интерфейсе пользователя что-то изменится, то TS подскажет нам, что и в классе необходимо что-то изменить.

Имплементацию можно проводить сразу от **несколько интерфейсов**:

```
1 interface Validation {  
2     valid: boolean;  
3     isValid: (data: string) => boolean;  
4 }  
5  
6 class UserForm implements IUser, Validation {  
7     login: string;  
8     password: string;  
9     valid: boolean = false;  
10  
11     isValid(login: string) {  
12         return login.length > 3;  
13     }  
14 }
```

Если не указать тип у аргумента login, то будет ошибка. Все потому, что класс **автоматически не перенимает** аннотации аргументов или все необязательные свойства интерфейса(ов)! Вы **сами** должны добавить аннотации типов и необязательные свойства по необходимости