



# Fluent Interface

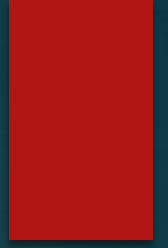
ВЫПОЛНИЛ СТУДЕНТ ГРУППЫ Б07-021-1

ЯКИМОВ ПАВЕЛ

# Текущий интерфейс

(англ. fluent interface) в разработке программного обеспечения – способ реализации объектно-ориентированного API, нацеленный на повышение читабельности исходного кода программы. Название придумано Эриком Эвансом и Мартином Фаулером.

# Передаваемый контекст



- ▶ определен с помощью значения, возвращаемого методом;
- ▶ наследуется (в качестве нового контекста используется предыдущий);
- ▶ прекращается возвращением ничего не значащего значения (void).

# Пример

```
var mark = db.Mark
    .Where(v => v.StudentId == student.Id)
    .Where(v => v.DisciplinId == disc.Id)
    .Where(v => v.SemestrId == semester.Id)
    .FirstOrDefault();
```



# Пример без использования Fluent Interface

```
var mark = db.Mark;  
mark.Where(v => v.StudentId == student.Id)  
mark.Where(v => v.DisciplineId == disc.Id)  
mark.Where(v => v.SemesterId == semester.Id)  
mark.FirstOrDefault();
```

# Продвинутое использование (пример из PHP)

```
$books = BookQuery::create()  
    ->useAuthorQuery()  
        ->filterByName('Leo Tolstoi')  
    ->endUse()  
    ->find();
```

# Пример без использования Fluent Interface

```
$authorQuery = AuthorQuery::create();  
$authorQuery->filterByName('Leo Tolstoi');  
$bookQuery = BookQuery::create();  
$bookQuery->setAuthorQuery($authorQuery);  
$books = $bookQuery->find();
```