

# Конфигурация с помощью аннотаций. @Autowired.

Возможно осуществлять конфигурацию контейнеров в Spring через аннотации.

Для это необходимо в конфигурации контейнера указать

```
<context:annotation-config/>
```

Именно её используют чаще всего в реальных проектах

**@Autowired** используется для автоматического связывания зависимостей.

Можно использовать с :

- сеттерами
- конструкторами
- методами с любым количеством аргументов
- Properties (свойства)
- Массивами и типизированными коллекциями
- В том числе и с приватными.

- Если имеется несколько бинов одинакового типа, чтобы задать конкретный бин используется **@Qualifier("имя")**, тогда будет связан бин с соответствующим ID.
- По умолчанию, если не найден подходящий бин, выбрасывается исключение. Это поведение можно изменить выставив **@Autowired(required=false)**.
- Только один конструктор данного типа бина может содержать аннотацию **@Autowired**, указывая, конструктор для связывания при использовании бина. Начиная со Spring 4.0 аннотацию **@Autowired** можно не ставить на конструктор, если он единственный в классе.
- Поля (помеченные **@Autowired** аннотацией) внедряются сразу после создания бина, до вызова каких-либо методов. Сеттеры вызываются последними.

# Разбор примера

- Класс Person содержит поле name, его сеттер и поле cat.
- Класс Cat – поле name.
- В контейнере содержатся бины person, cat и cat1.

- Запустите main: бин cat автоматически свяжется.

Вывод: Person{name='Ivan', cat=Cat{name='Barsik'}}

- Как измениться вывод, если закоментировать бин cat и раскомментировать cat1?

(Person{name='Ivan', cat=Cat{name='Vas'ka'}})

- Раскомментируйте бин cat и @Qualifier у сеттера. Теперь снова связывается бин cat, т.к. он указан в @Qualifier.
- Раскомментируйте

```
// @Autowired  
// @Qualifier("cat1")  
// public void prepare(Cat cat){  
//     System.out.println(cat);  
// }
```

Проверьте вывод. Каждый метод связался с указанным через @Qualifier бином.