


Autowiring - автоматическое связывание.

Спринг способен определять и добавлять зависимости автоматически, что может значительно уменьшить объём конфигураций(написанного кода), но в то же время делает затруднительнее чтение и проверку зависимостей.

Существует три типа автосвязывания:

- `byName`: контейнер ищет бин с ID идентичным указанному. Если такой бин не находится, то связывания не происходит.
- `byType`: контейнер ищет бин определённого класса, работает если существует только 1 бин типа, указанного в `property`, иначе выбрасывается `UnsatisfiedDependencyException`.

- constructor: создаёт объект, используя конструктор, использует byType режим.
- Для использования автосвязывания надо указать в бине  И указать тип связывания.

В нашем примере имеем

- интерфейс Animal (животное), три класса его реализующие: Cat, Dog и Parrot(кот, собака, попугай)
- Три класса – содержащих в себе эти бины (AnimalShelter, CatPerson, PetPerson).
- Два файла с контекстами – с автосвязыванием и без него (задаём зависимости старым добрым ref).

(зачем hashCode() в main? И зачем попугай, если его нигде не юзаем)

- Сравните XML файлы с использованием автосвязывания и без.
 - Запустите программу с обоими вариантами.
1. В XML файле с использованием автосвязывания раскомментируйте бин `animal1` и оставьте бин `animal`. Выбросится ошибка. Почему?
 2. Если раскомментировать этот бин `animal1` и закомментировать `animal` бин, то автосвязывания `byName` не произойдёт. Почему?

1. Оба бина одного типа Cat, поэтому автосвязывание по типу невозможно.
2. Имена «animal1» (бина класса Cat) и зависимости бина animalShelter «animal» - разные, поэтому связывания не произойдёт.

```
public class PetPerson {  
    Animal animal;
```

```
<bean id="petPerson"  
      class="com.volkov.IoC.example13.PetPerson"  
      autowire="byName">  
</bean>
```

При создании класса petPerson, контейнер «смотрит» на класс(тип) поля (в нашем случае Animal), находит animal бин, и подставляет его по имени.

Т.е. если мы поменяем имя переменной на animal1

```
public class PetPerson {  
    Animal animal1;  
}
```

```
<bean id="petPerson"  
      class="com.volkov.ioC.example13.PetPerson"  
      autowire="byName">  
</bean>
```

Связывания всё равно не произойдёт, потому что имя = названию класса (только с маленькой буквы).