Реализация DI-контейнера с минимальной конфигурацией для языка Java

Денис Мухаметьянов

20 июня 2013

Код без соблюдения принципа инверсии зависимостей

```
public A()
{
    this.b = new B();
    this.c = new C();
}
```

Код с соблюдением принципа инверсии зависимостей

```
public A(IB b, IC c)
{
    this.b = b;
    this.c = c;
}
```

Код без соблюдения принципа инверсии зависимостей

```
public A()
{
    this.b = new B();
    this.c = new C();
}
```

Код с соблюдением принципа инверсии зависимостей

```
public A(IB b, IC c)
{
    this.b = b;
    this.c = c;
}
```

Код без соблюдения принципа инверсии зависимостей

```
public A()
{
    this.b = new B();
    this.c = new C();
}
```

Код с соблюдением принципа инверсии зависимостей

```
public A(IB b, IC c)
{
    this.b = b;
    this.c = c;
}
```

Наблюдение

Должно быть место, где создаются и хранятся объекты.

ServiceLocator

```
ServiceLocator.register(IB.class, new B());
ServiceLocator.register(IC.class, new C());
ServiceLocator.register(IA.class, new A(ServiceLocator.get(
    IB.class), ServiceLocator.get(IC.class)));
```

Наблюдение

Должно быть место, где создаются и хранятся объекты.

ServiceLocator

```
ServiceLocator.register(IB.class, new B());
ServiceLocator.register(IC.class, new C());
ServiceLocator.register(IA.class, new A(ServiceLocator.get(
    IB.class), ServiceLocator.get(IC.class)));
```

Решение проблем ServiceLocator'а

DI-контейнер берет на себя ответственность за создание объектов

Spring IoC Container

```
<bean id="AClassImpl" class="A" />
<bean id="BClassImpl" class="B" />
<bean id="CClassImpl" class="C" />
```

Google Guice

```
bind(IA.class).to(A.class);
bind(IB.class).to(B.class);
bind(IC.class).to(C.class);
```

Spring IoC Container

```
<bean id="AClassImpl" class="A" />
<bean id="BClassImpl" class="B" />
<bean id="CClassImpl" class="C" />
```

Google Guice

```
bind(IA.class).to(A.class);
bind(IB.class).to(B.class);
bind(IC.class).to(C.class);
```

Spring IoC Container

```
<bean id="AClassImpl" class="A" />
<bean id="BClassImpl" class="B" />
<bean id="CClassImpl" class="C" />
```

Google Guice

```
bind(IA.class).to(A.class);
bind(IB.class).to(B.class);
bind(IC.class).to(C.class);
```

Можно ли реализовать DI-контейнер, конфигурирование которого не потребует большого количества кода?

JRoboContainer

Разработан JRoboContainer — контейнер, который позволяет избежать явного перечисления всех классов

Можно ли реализовать DI-контейнер, конфигурирование которого не потребует большого количества кода?

JRoboContainer

Разработан JRoboContainer — контейнер, который позволяет избежать явного перечисления всех классов

Контейнер готов к использованию сразу после создания

Имеется возможность точечной настройки

- аннотация @ContainerConstructor
- указание конкретной реализации абстракции
- указание конкретного экземпляра абстракции
- указание конкретного загрузчика класса

Контейнер готов к использованию сразу после создания

Имеется возможность точечной настройки

- аннотация @ContainerConstructor
- указание конкретной реализации абстракции
- указание конкретного экземпляра абстракции
- указание конкретного загрузчика класса

Контейнер готов к использованию сразу после создания

Имеется возможность точечной настройки

- аннотация @ContainerConstructor
- указание конкретной реализации абстракции
- указание конкретного экземпляра абстракции
- указание конкретного загрузчика класса

Контейнер готов к использованию сразу после создания

Имеется возможность точечной настройки

- аннотация @ContainerConstructor
- указание конкретной реализации абстракции
- указание конкретного экземпляра абстракции
- указание конкретного загрузчика класса

Контейнер готов к использованию сразу после создания

Имеется возможность точечной настройки

- аннотация @ContainerConstructor
- указание конкретной реализации абстракции
- указание конкретного экземпляра абстракции
- указание конкретного загрузчика класса

Контейнер готов к использованию сразу после создания

Имеется возможность точечной настройки

- аннотация @ContainerConstructor
- указание конкретной реализации абстракции
- указание конкретного экземпляра абстракции
- указание конкретного загрузчика класса

- Код тщательно протестирован
- Контейнер внедрен в реальный проект

- Код тщательно протестирован
- Контейнер внедрен в реальный проект

https://code.google.com/p/jrobo-container/