

Using the Jakarta Commons

Cli (Command Line Interface)

Tek bir standart arabirimle, komut satırından parametre almak için.

Örnek:

```
java CLIDemo -n Mert Nuhoglu -t 16:37
```

Çıktı:

```
Date/Time: Wed Mar 10 16:38:55 EET 2004  
Nice to meet you: Mert
```

Bu tarz, program parametrelerini belirtebiliyoruz.

```
options.addOption("n", true, "[name] your name");  
  
Option timeOption = new Option("t", false, "current time");
```

Discovery

Sınıfları ve kaynakları tespit etmek için.

```
// 1. Use the supplied ClassLoaders  
ClassLoaders loaders =  
    ClassLoaders.getAppLoaders(MyInterface.class, getClass(), false);  
  
// 2. DiscoverClass is used to find our implementing classes  
DiscoverClass discover = new DiscoverClass(loaders);  
  
// 3. find the implementing class  
Class implClass = discover.find(MyInterface.class);
```

MyInterface interfaceinin hangi implementasyonunu kullanacağımızı sonraya bıraktık. Kullanacağımız implementasyonu ya META-INF/services altındaki bir dosyada belirtiyoruz. Ya da system propertyler kullanıyoruz:

```
$ java -DMyInterface=MyImpl1 DiscoveryDemo
```

Böylece belirli bir interfacein hangi **implementasyonunu** kullanacağımızı runtime'da tespit ederiz.

Lang

String işlemleri için faydalı metotlar.

C benzeri enumlar.

Örnek:

```
StringUtils.capitalize("name");
```

RandomStringUtils, rassal metinler üretir.

NumberUtils, sayıların maksimumu, minimumu, sayıları metine çevirmek.

Örnek Kod:

```
// -- StringUtils methods  
System.err.println("==== StringUtils methods =====");  
  
// pads and centers a string with the specified string around it
```

```

System.err.println(StringUtils.center("name", 8, "Da"));

// removes everything from a string after the specified separator
System.err.println(StringUtils.chomp("TVGuide", "Gu"));

// returns how many times a substring occurs in the given string
System.err.println(StringUtils.countMatches("DumDeeDumDeeDumDaDa", "Dee"));

// deletes all white spaces from the given string
System.err.println(StringUtils.deleteWhitespace(" Whit e Spa ce "));

// replaces all occurrences of a string with the given string
System.err.println(StringUtils.replace("DaDaDumDee", "Dum", "Doo"));

// reverses a string
System.err.println(StringUtils.reverse("1234567"));

// -- RandomStringUtils methods
System.err.println("==== RandomStringUtils methods =====");

// creates a random string of specified length and contains numbers
System.err.println(RandomStringUtils.random(8, true, true));

// -- NumberUtils methods
System.err.println("==== NumberUtils methods =====");

// prints the maximum of the three numbers
System.err.println(NumberUtils.maximum(8, 5, -9));

// prints the minimum of the three numbers
System.err.println(NumberUtils.minimum(8, 5, -9));

// -- NumberRange methods
System.err.println("==== NumberRange methods =====");

// create a number range object with -10 as the min and 10 as the max
NumberRange nr = new NumberRange(new Integer(-10), new Integer(10));

// returns whether the number is included in this range
System.err.println("Includes 0? " + nr.includesNumber(new Integer(0)));

// -- Builder Methods - Require the Mortgage class
System.err.println("==== Builder methods =====");
Mortgage mortgage1 = new Mortgage();
mortgage1.setYears(25);
mortgage1.setRate(4.55f);

Mortgage mortgage2 = new Mortgage();
mortgage2.setYears(25);
mortgage2.setRate(4.56f);

// returns the toString representation of this mortgage class using '
// ToStringBuilder
System.err.println("Mortgage 1: " + mortgage1.toString());

// checks whether Mortgage1 is equal to Mortgage2 using ToEqualsBuilder
System.err.println("Is Mortgage 1 is equal to Mortgage 2? " +
    mortgage1.equals(mortgage2));

// -- Enum Methods - Require the OnTV class
System.err.println("==== Enum methods =====");

// grab a value from the Enum
System.err.println("From OnTV Enum, Idol item: " + OnTV.getEnum("Idol"));

```

Çıktı:

```

===== StringUtils methods =====
DanameDa
TVGuide
2
WhiteSpace
DaDaDooDee
7654321

```

```
===== RandomStringUtils methods =====
```

```
PaMWtRbW
```

```
===== NumberUtils methods =====
```

```
8
```

```
-9
```

```
===== NumberRange methods =====
```

```
Includes 0? true
```

```
===== Builder methods =====
```

```
Mortgage 1: Mortgage@1820dda[rate=4.55,years=25]
```

```
Is Mortgage 1 is equal to Mortgage 2? false
```

```
===== Enum methods =====
```

```
From OnTV Enum, Idol item: OnTV[Idol]
```

```
---
```

Hazır `toString`, `hashCode`, `compareTo` ve `equals` metotlarını Builder'a ürettirmeli.

```
        return new EqualsBuilder()
            .append(this.rate, m2.rate)
            .append(this.years, m2.years)
            .isEquals();
```

```
        return new ToStringBuilder(this)
            .append("rate", this.rate)
            .append("years", this.years)
            .toString();
```

```
---
```