

Test Spring el with ExpressionParser

By mkyong (<http://www.mkyong.com/author/mkyong/>) | June 14, 2011 | Updated : August 29, 2012

Spring expression language (SpEL) supports many functionality, and you can test those expression features with this special “**ExpressionParser**” interface.

Here’s two code snippets, show the basic usage of using Spring EL.

SpEL to evaluate the literal string expression.

```
ExpressionParser parser = new SpelExpressionParser();
Expression exp = parser.parseExpression("'put spel expression here'");
String msg = exp.getValue(String.class);
```

SpEL to evaluate the bean property – “item.name”.

```
Item item = new Item("mkyong", 100);
StandardEvaluationContext itemContext = new StandardEvaluationContext(item);

//display the value of item.name property
Expression exp = parser.parseExpression("name");
String msg = exp.getValue(itemContext, String.class);
```

Few examples to test SpEL. The codes and comments should be self-exploratory.

```
import org.springframework.expression.Expression;
import org.springframework.expression.ExpressionParser;
import org.springframework.expression.spel.standard.SpelExpressionParser;
import org.springframework.expression.spel.support.StandardEvaluationContext;

public class App {
    public static void main(String[] args) {

        ExpressionParser parser = new SpelExpressionParser();

        //literal expressions
        Expression exp = parser.parseExpression("'Hello World'");
        String msg1 = exp.getValue(String.class);
        System.out.println(msg1);

        //method invocation
        Expression exp2 = parser.parseExpression("'Hello World'.length()");
        int msg2 = (Integer) exp2.getValue();
        System.out.println(msg2);

        //Mathematical operators
        Expression exp3 = parser.parseExpression("100 * 2");
        int msg3 = (Integer) exp3.getValue();
        System.out.println(msg3);

        //create an item object
        Item item = new Item("mkyong", 100);
        //test EL with item object
        StandardEvaluationContext itemContext = new StandardEvaluationContext(item);

        //display the value of item.name property
        Expression exp4 = parser.parseExpression("name");
        String msg4 = exp4.getValue(itemContext, String.class);
        System.out.println(msg4);

        //test if item.name == 'mkyong'
        Expression exp5 = parser.parseExpression("name == 'mkyong'");
        boolean msg5 = exp5.getValue(itemContext, Boolean.class);
        System.out.println(msg5);

    }
}
```

```
public class Item {  
  
    private String name;  
  
    private int qty;  
  
    public Item(String name, int qty) {  
        super();  
        this.name = name;  
        this.qty = qty;  
    }  
  
    //...  
}
```

Output

```
Hello World  
11  
200  
mkyong  
true
```

Note

This article is demonstrates few basic usages of Spring expression parser, and you should visit this official Spring expression documentation (<http://static.springsource.org/spring/docs/3.0.x/spring-framework-reference/html/expressions.html>) for hundred of useful SpEL examples.

Download Source Code

Download It – Spring3-EL-Parser-Example.zip (<http://www.mkyong.com/wp-content/uploads/2011/06/Spring3-EL-Parser-Example.zip>) (6 KB)

Tags : [spring el](http://www.mkyong.com/tag/spring-el/) (<http://www.mkyong.com/tag/spring-el/>)
[spring3](http://www.mkyong.com/tag/spring3/) (<http://www.mkyong.com/tag/spring3/>)

Share this article on

Twitter (<https://twitter.com/intent/tweet?text=Test Spring el with ExpressionParser&url=http://www.mkyong.com/spring3/test-spring-el-with-expressionparser/&via=mkyong>) Facebook (<https://www.facebook.com/sharer/sharer.php?u=http://www.mkyong.com/spring3/test-spring-el-with-expressionparser/>) Google+