



链滴

Java8 特性详解 --Streams API 实战之 分解质因数

作者: [washmore](#)

原文链接: <https://ld246.com/article/1504748460342>

来源网站: [链滴](#)

许可协议: [署名-相同方式共享 4.0 国际 \(CC BY-SA 4.0\)](#)

<p>**刚看到群里有个人提问 100 如何分解为 $2*2*5*5$ 的形式...于是随手丢了一个 demo 出来 **</p>

<p>直接上源码,过程详见注释:</p>

```
<pre><code class="highlight-chroma"><span class="highlight-line"><span class="highlight-cl">@Test
</span></span><span class="highlight-line"><span class="highlight-cl">public void test() {
</span></span><span class="highlight-line"><span class="highlight-cl">    //原始数据
</span></span><span class="highlight-line"><span class="highlight-cl">    int source = 100;
</span></span><span class="highlight-line"><span class="highlight-cl">    //质因数合集
</span></span><span class="highlight-line"><span class="highlight-cl">    List list = new Ar
ayList<&lt;&gt;&gt;();
</span></span><span class="highlight-line"><span class="highlight-cl">    //分解质因数
</span></span><span class="highlight-line"><span class="highlight-cl">    memeda(list, sou
ce);
</span></span><span class="highlight-line"><span class="highlight-cl">    //输出质因数合

</span></span><span class="highlight-line"><span class="highlight-cl">    System.out.print
n(list);
</span></span><span class="highlight-line"><span class="highlight-cl">    //还原质因数乘

</span></span><span class="highlight-line"><span class="highlight-cl">    int target = list.st
eam().reduce((t, i) -&gt; t * i).get();
</span></span><span class="highlight-line"><span class="highlight-cl">    //验证结论
</span></span><span class="highlight-line"><span class="highlight-cl">    Assert.assertEqua
s(source, target);
</span></span><span class="highlight-line"><span class="highlight-cl">}</span></span>
</span></span><span class="highlight-line"><span class="highlight-cl"><span class="highlight-line"><span class="highlight-cl">
</span></span><span class="highlight-line"><span class="highlight-cl">public static void
memeda(List children, int source) {
</span></span><span class="highlight-line"><span class="highlight-cl">    if (source &gt; 1
{
</span></span><span class="highlight-line"><span class="highlight-cl">        children.add(
Optional
ofNullable(
</span></span><span class="highlight-line"><span class="highlight-cl">            Int
tream.range(2, source)
</span></span><span class="highlight-line"><span class="highlight-cl">            .filter(i -&gt; source % i == 0)
</span></span><span class="highlight-line"><span class="highlight-cl">            .findFirst()
</span></span><span class="highlight-line"><span class="highlight-cl">            ).get().o
Else(source)
</span></span><span class="highlight-line"><span class="highlight-cl">        });
</span></span><span class="highlight-line"><span class="highlight-cl">        memeda(chil
ren, source / children.get(children.size() - 1));
</span></span><span class="highlight-line"><span class="highlight-cl">    }
</span></span><span class="highlight-line"><span class="highlight-cl"></span></span><span class="highlight-line"><span class="highlight-cl"></span></span></code></pre>
<p>输出内容为:</p>


```
<pre><code class="highlight-chroma">[2, 2, 5, 5]
1 test passed
</code></pre>
<p>本例使用 java8 的 stream api,就当随手笔记了...</p>
```


```