## 5. 生成器（Builder）

### Intent

封装一个对象的构造过程，并允许按步骤构造。

### Class Diagram

### Implementation

以下是一个简易的 StringBuilder 实现，参考了 JDK 1.8 源码。

public class AbstractStringBuilder {
 protected char[] value;

 protected int count;

 public AbstractStringBuilder(int capacity) {
 count = 0;
 value = new char[capacity];
 }

 public AbstractStringBuilder append(char c) {
 ensureCapacityInternal(count + 1);
 value[count++] = c;
 return this;
 }

 private void ensureCapacityInternal(int minimumCapacity) {
 // overflow-conscious code
 if (minimumCapacity - value.length > 0)
 expandCapacity(minimumCapacity);
 }

 void expandCapacity(int minimumCapacity) {
 int newCapacity = value.length \* 2 + 2;
 if (newCapacity - minimumCapacity < 0)
 newCapacity = minimumCapacity;
 if (newCapacity < 0) {
 if (minimumCapacity < 0) // overflow
 throw new OutOfMemoryError();
 newCapacity = Integer.MAX\_VALUE;
 }
 value = Arrays.copyOf(value, newCapacity);
 }
}

public class StringBuilder extends AbstractStringBuilder {
 public StringBuilder() {
 super(16);
 }

 @Override
 public String toString() {
 // Create a copy, don't share the array
 return new String(value, 0, count);
 }
}

public class Client {
 public static void main(String[] args) {
 StringBuilder sb = new StringBuilder();
 final int count = 26;
 for (int i = 0; i < count; i++) {
 sb.append((char) ('a' + i));
 }
 System.out.println(sb.toString());
 }
}

abcdefghijklmnopqrstuvwxyz

### JDK

* [java.lang.StringBuilder](http://docs.oracle.com/javase/8/docs/api/java/lang/StringBuilder.html)
* [java.nio.ByteBuffer](http://docs.oracle.com/javase/8/docs/api/java/nio/ByteBuffer.html#put-byte-)
* [java.lang.StringBuffer](http://docs.oracle.com/javase/8/docs/api/java/lang/StringBuffer.html#append-boolean-)
* [java.lang.Appendable](http://docs.oracle.com/javase/8/docs/api/java/lang/Appendable.html)
* [Apache Camel builders](https://github.com/apache/camel/tree/0e195428ee04531be27a0b659005e3aa8d159d23/camel-core/src/main/java/org/apache/camel/builder)