

```

package util;

/** A class that represents an array of booleans on the bit level. */
public class BooleanArray {
    private byte[] values; // The array that holds the data
    private int size; // The number of bits in the array

    // An array that holds the values
    private static final byte bitIndices[] = {(byte)0x01, (byte)0x02, (byte)0x04, (byte)0x08, (byte)
)0x10, (byte)0x20, (byte)0x40, (byte)0x80};
    private static final byte all = (byte)0xff/* All bits true. */, nothing = (byte)0x00; /* All bi
ts false. */
    /** Initializes a BooleanArray of capacity equal to the given number
    * rounded up to a multiple of eight with all values initialized to the
    * given boolean value. */
    private BooleanArray(int capacity, boolean initialValues){
        initializeArray(capacity, initialValues);
    }

    /** Does all the actual work of initialization. */
    private void initializeArray(int capacity, boolean initVal){
        this.size = capacity + ((capacity%8 != 0) ? (8-(capacity%8)) : 0);
        this.values = new byte[this.size/8];
        byte init = initVal ? all : nothing;
        for(int i=0; i<this.size; i++){
            this.values[i/8] = init;
        }
    }

    /** Initializes a BooleanArray of capacity equal to the given number
    * rounded up to a multiple of eight with all values initialized to false. */
    public static BooleanArray create(int capacity){
        return new BooleanArray(capacity, false);
    }

    /** Initializes a BooleanArray of capacity equal to the given number
    * rounded up to a multiple of eight with all values initialized to the
    * given boolean value. */
    public static BooleanArray create(int capacity, boolean initialValues){
        return new BooleanArray(capacity, initialValues);
    }

    /** Gets the boolean value stored at the given index. Index must be
    * smaller than the capacity as retrieved by getCapacity. */
    public boolean get(int index){
        return (this.values[index/8] & bitIndices[index%8]) == bitIndices[index%8];
    }

    /** Sets the boolean value stored at the given index to the given value.
    * Index must be smaller than the capacity as retrieved by getCapacity. */
    public void set(int index, boolean val){
        this.values[index/8] = (byte) (val ? this.values[index/8] | bitIndices[index%8] : this.
values[index/8] & (~bitIndices[index%8]));
    }

    /** Sets the boolean value stored at the given index to true.
    * Index must be smaller than the capacity as retrieved by getCapacity. */
    public void setTrue(int index){
        this.values[index/8] = (byte) (this.values[index/8] | bitIndices[index%8]);
    }

    /** Sets the boolean value stored at the given index to false.
    * Index must be smaller than the capacity as retrieved by getCapacity. */
    public void setFalse(int index){
        this.values[index/8] = (byte) (this.values[index/8] & (~bitIndices[index%8]));
    }

    /** Gets the capacity of the BooleanArray. */
    public int getCapacity(){
        return size;
    }
}

```