NOTE: SCENARIO IS WHAT THE PUZZLE TAKER SEES.

SCENARIO:

See the J24-cipher-streams-close puzzle.

Code:

```
// OMITTED: Import whatever is needed.
public final class CryptoUtils {
  public static byte[] encrypt (Cipher cipher, String text)
      throws IOException {
    ByteArrayOutputStream bytes = new ByteArrayOutputStream();
    InputStream input = new ByteArrayInputStream(text.getBytes());
    OutputStream processor = new CipherOutputStream(bytes, cipher);

    // IOUtils.copy(input, processor): Reads all bytes from
    // 'input' stream and writes them to 'processor' stream.
    IOUtils.copy(input, processor);

    processor.close();
    return bytes.toByteArray();
  }
}
```

Questions:
See the J24-cipher-streams-close puzzle.

NOTE: ANSWER IS TO BE SHOWN TO THE PUZZLE TAKER AT THE END OF THE SESSION.

ANSWER:
a
See the J24-cipher-streams-close puzzle.

NOTE: THE REST OF THE DOCUMENT CONTAINS EXTRA INFORMATION FOR THE PROJECT RESEARCHERS. IT IS NOT TO BE SHOWN TO THE PUZZLE TAKERS.

TAGS:
java, cryptography, cipher, invalid-object-initialization, api-protocol-usage,

CATEGORIES:
Blindspot - YES
Type - Crypto

Number of distinct functions - 6
Number of total functions - 6
Blindspot function - NA
Function call omitted - NA
Blindspot type - NA
Number of Parameters in the blindspot function - NA
Cyclomatic complexity - 2

NAME:
CipherInputStream, CipherOutputStream - Filter streams to process (read or write) the data before doing the main IO operations (read and write)

MORE INFORMATION:
To see an incorrect way to use the API, look at the J24-cipher-streams-close puzzle.

REFERENCES:

1. CipherInputStream#close
2. CipherOutputStream#close