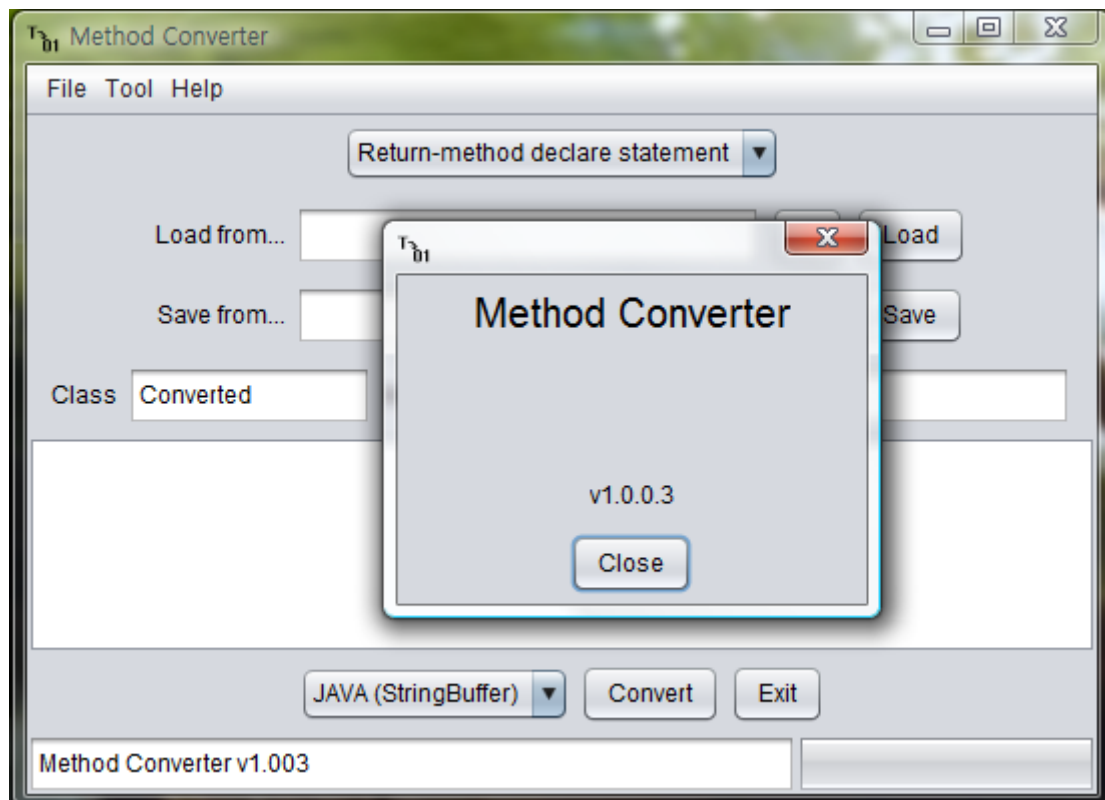


METHOD CONVERTER

User Manual



Made by HJOW

hujinone22@naver.com

Installation & Run

You need to install JAVA to run this program.

Visit <http://java.com/download/> and download JAVA.

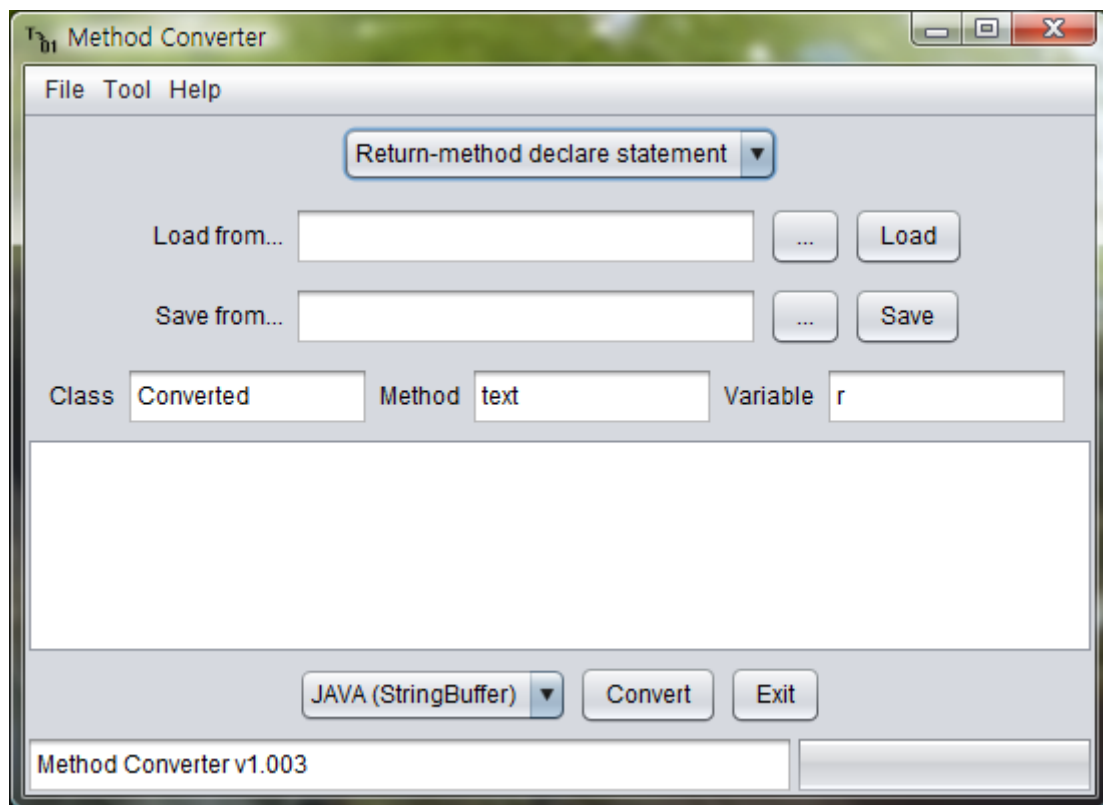
And then, just run **methodConverter.bat** to execute this program.

If you use another OS, open terminal and use commands

```
java -jar methodConverter.jar
```

You should write full path of *methodConverter.jar*.

Interface



There is a menu bar on the top of the window.

You can use whole functions with using menu.

There is mode-selector under the menu bar.

You can select what you want, return-method-declare-statement, text encryption, etc.

There are several buttons under the mode-selector.

You can load, or save text files.

There is a center area. Loaded text will be shown here. Also, converted text will be shown here.

You can edit text here. When the convert button is pressed, text which is in center area will be converted, and then, will be shown at the center area.

Return-method-declare-statement Example

This function help you to embed the text into the source code.

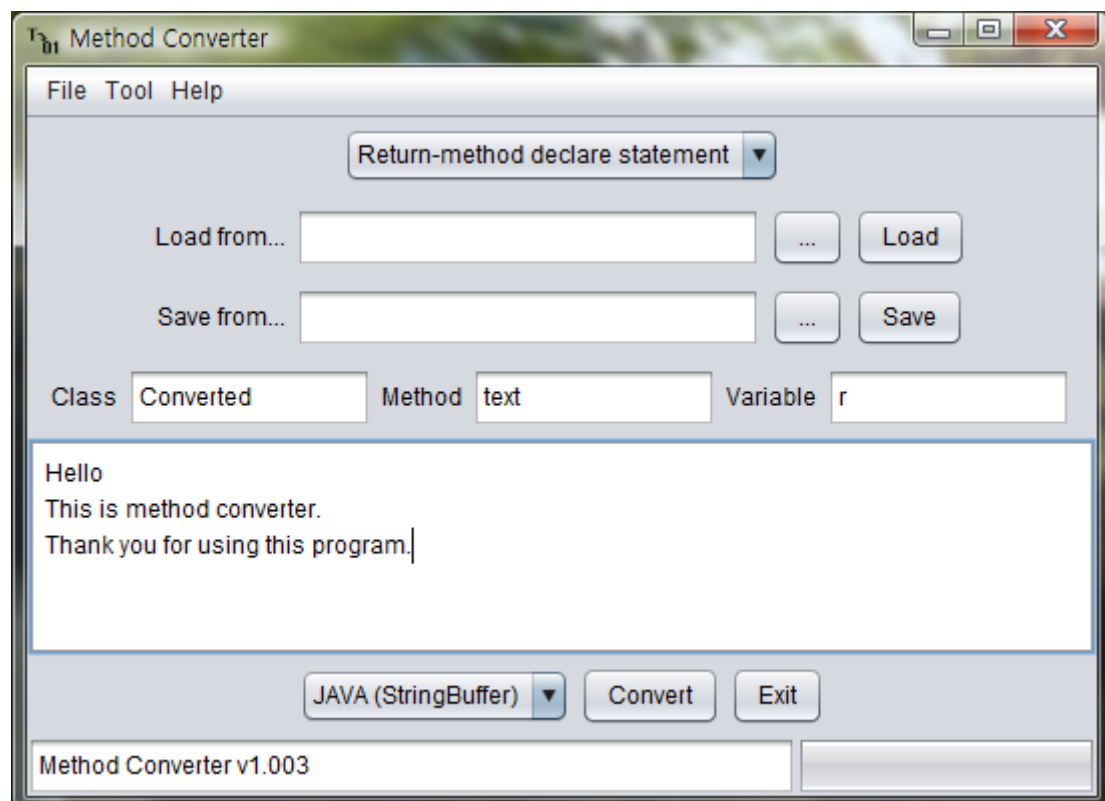
1. Input some text in the center area.

In this example, I use following text.

Hello

This is method converter.

Thank you for using this program.



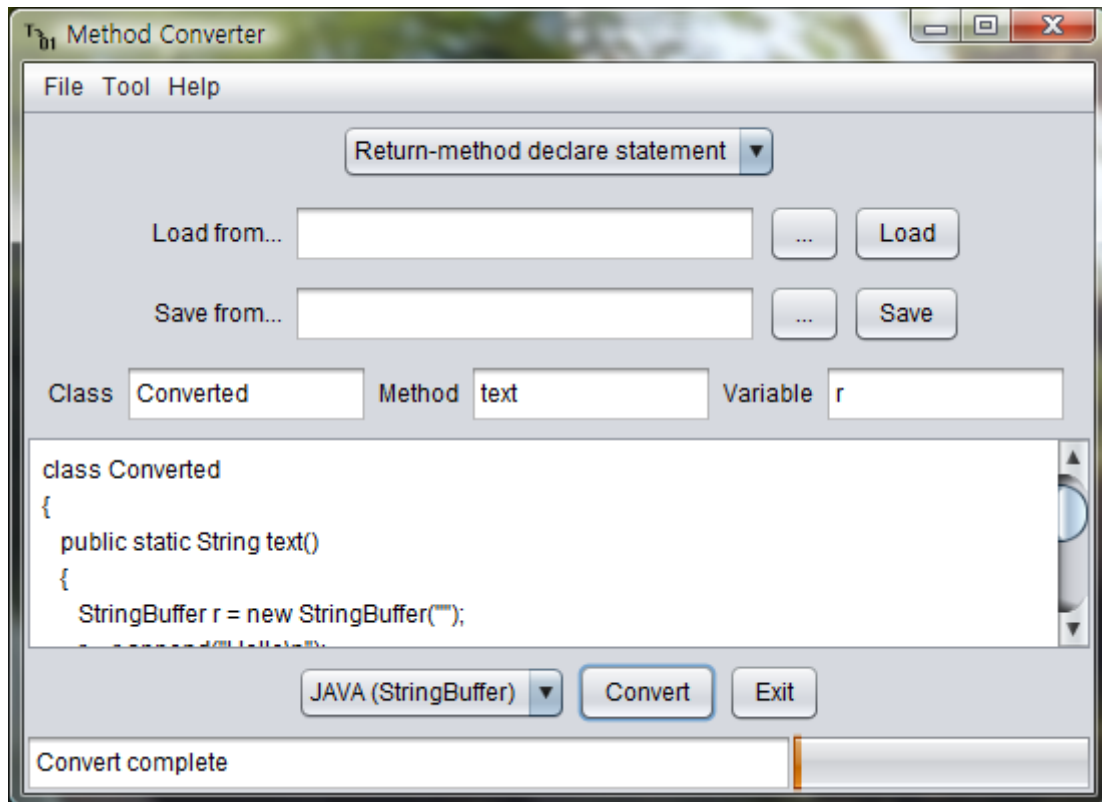
2. Input class, method, and variable name.

In this example, I use "Converted", "text", and "r".

3. Select programming-language syntax.

In this example, I use "JAVA (StringBuffer)".

4. Press the convert button.

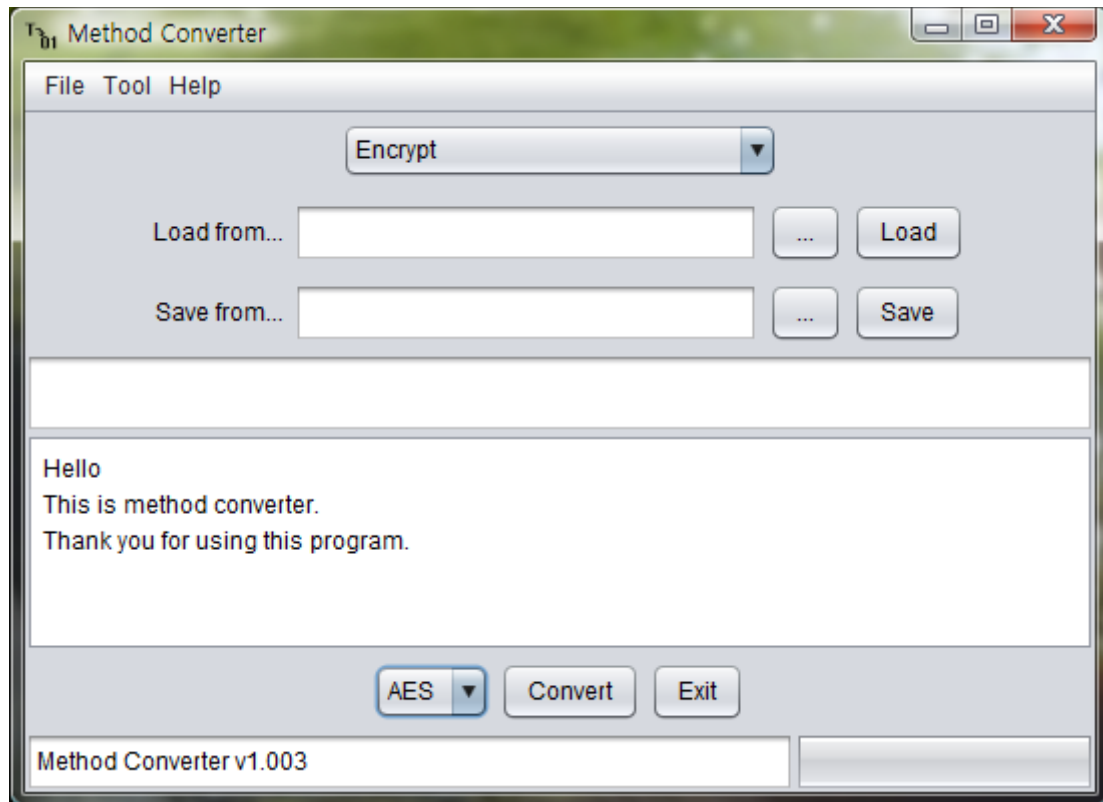


And then, just use the results.

```
class Converted
{
    public static String text()
    {
        StringBuffer r = new StringBuffer("");
        r = r.append("Hello\n");
        r = r.append("This is method converter.\n");
        r = r.append("Thank you for using this program.\n");
        return r;
    }
}
```

Encrypting Example

1. Input the text you want to encrypt.



In this example, I use following text.

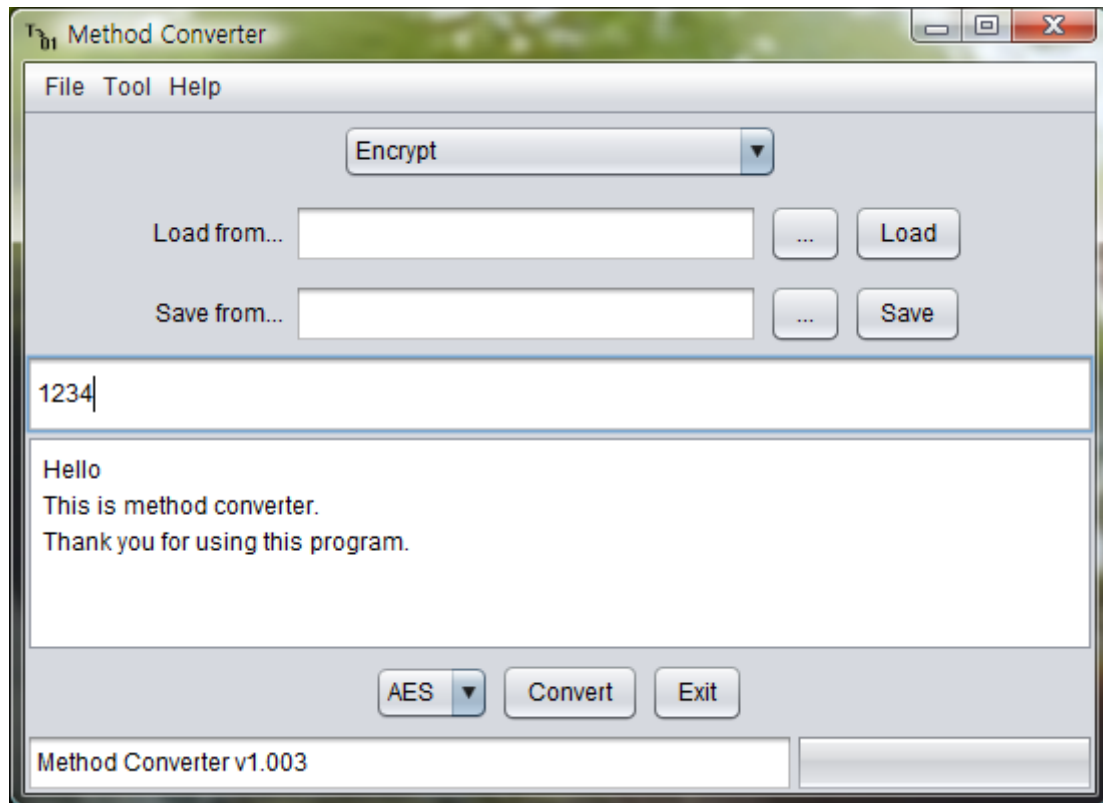
Hello

This is method converter.

Thank you for using this program.

2. Input the password.

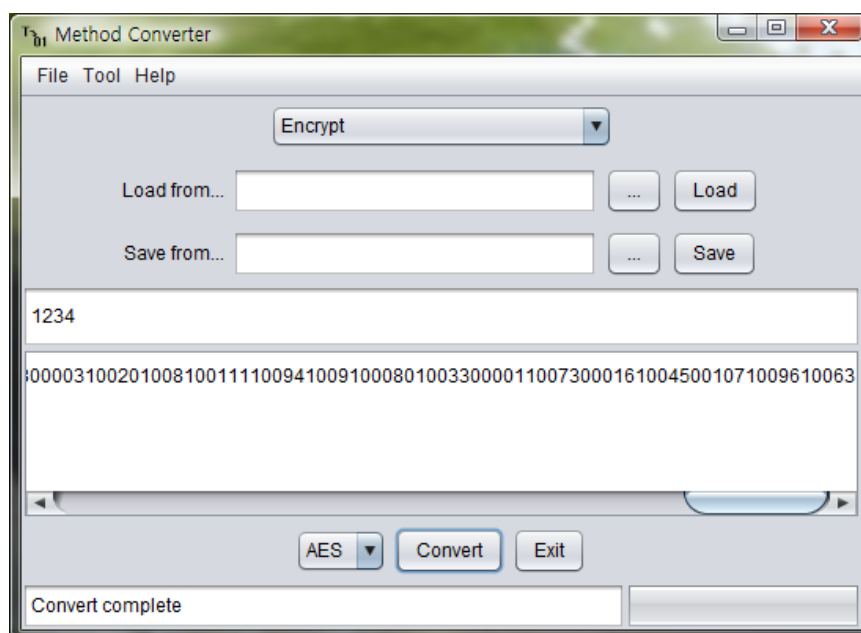
In this example, I use "1234".



3. Select the encrypting method.

In this example, I use *AES*.

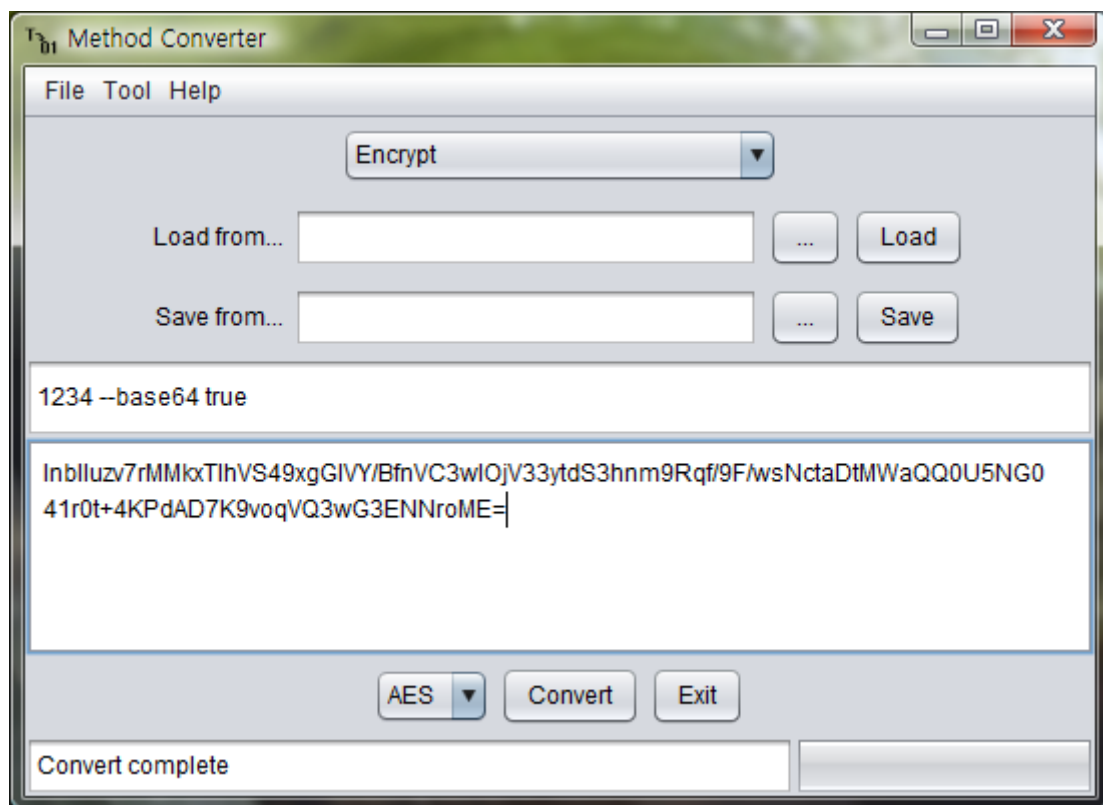
4. Press the convert button.



And then, just use the results.

101060011810056000341002010017100181007700012101090002010056101230008410072100090
002400006000330008600063000051000710043000110012400037000580005300119100330004300
093000750012000103101011004400106001271000100069100010001100013001141004210125100
761005910102000650001300020100281004710076100290009010012100731001800010000611004
8000031002010081001111009410091000801003300001100730001610045001071009610063

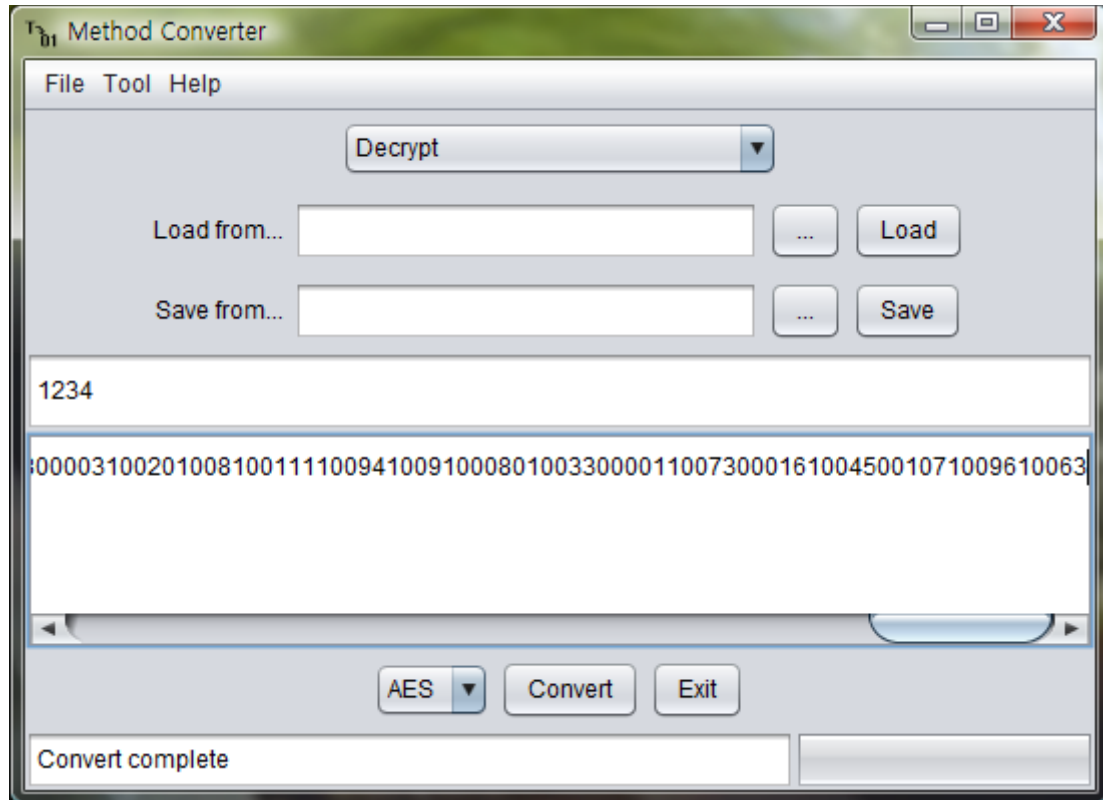
If you want to use BASE64 instead of these long-code, input the password with additional parameters "*base64*".



Decrypting Example

1. Input the text you want to encrypt.

Input what you want to decrypt.



In this example, I use following text.

```
101060011810056000341002010017100181007700012101090002010056101230008410072100090
002400006000330008600063000051000710043000110012400037000580005300119100330004300
093000750012000103101011004400106001271000100069100010001100013001141004210125100
761005910102000650001300020100281004710076100290009010012100731001800010000611004
8000031002010081001111009410091000801003300001100730001610045001071009610063
```

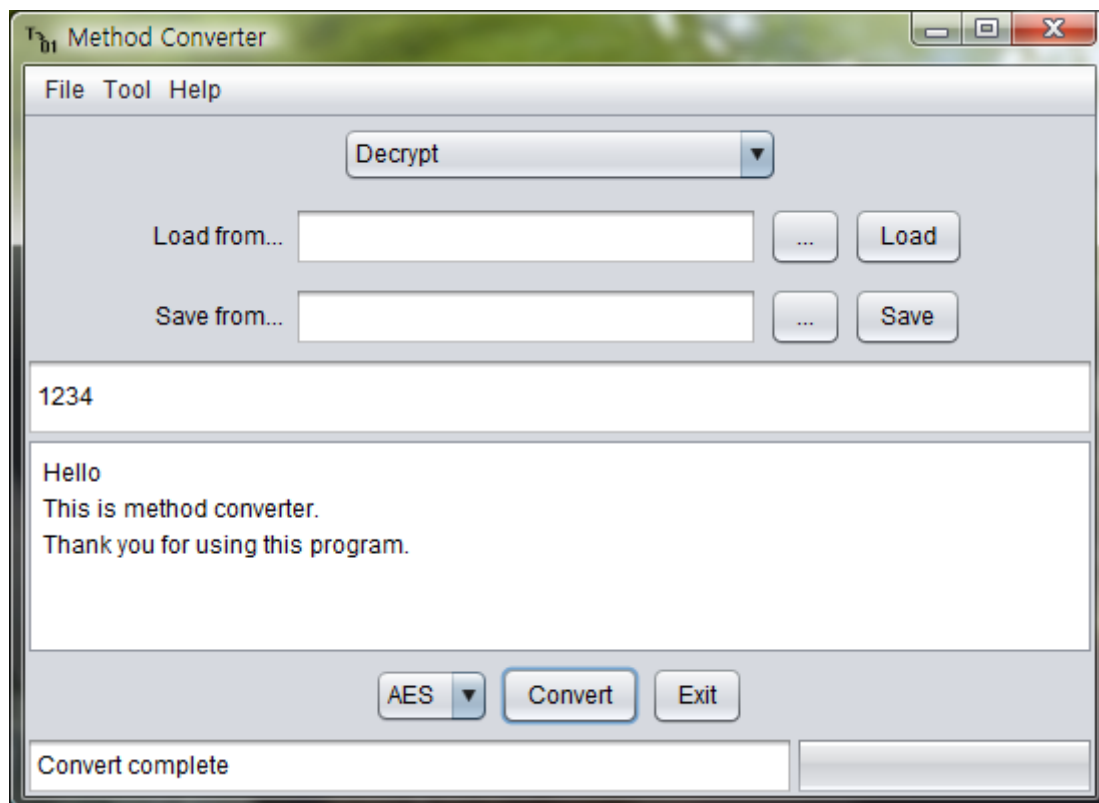
This is encrypted text at the encrypting example.

Let's see this text can be restored successfully.

2. Input the password, and select the decrypting method.

You should select correct password and method to restore original text successfully !

3. Press the convert button



You can see the original text.

License

You can use this program as commercial, and non-commercial use for free.

Also, you can use the **jar** file as library.

But, if you want to modify the source code of this program and publish as a new program, you should rename your program, or you should open your modified source code.