

Inderjeet,  
Nishant Pratap Savita,  
Gunjan

|  |
| --- |
| Code Encrypter |
| Authored by: |
| December 8, 2023 |



Code Encrypter

### Introduction

Protecting source code from unauthorized use and theft is critical for software developers in today's digital landscape. We have developed an innovative code encryption solution that transforms source code into an encrypted format that is completely unreadable and secure from prying eyes. At the same time, the encrypted code remains fully functional and can be compiled and executed just like normal code. This enables secure collaboration on software projects without revealing any sensitive logic or intellectual property in the source code. With our code encrypter, developers can finally share and work on code securely without compromising functionality or productivity. This solution represents a major advancement in keeping source code integrity and intellectual property safe in the face of growing cyber threats.

### Why Use Code Encrypter?

* **Protection from Code Theft:** In an era where code is a valuable commodity, protecting it from theft or unauthorized access is paramount. The Code Encrypter ensures that your code remains shielded from potential threats.
* **Maintain Code Functionality**: One of the standout features of the Code Encrypter is that the encrypted code remains functional. This means that while the code's readability is compromised, its execution is not hampered.
* **Ease of Use**: With a user-friendly interface and straightforward operations, the Code Encrypter is accessible to both novices and seasoned developers alike.
* **Provides Offline Encryption for Enhanced Security:** The encryption process happens entirely offline, isolated from external networks. This removes potential vulnerabilities of data leakage or remote attacks, keeping your source code protection air-gapped for maximum security.
* **Requires No Infrastructure or Network Access:** The solution is standalone and does not require setup of credentials or accounts. Just encrypt and share offline.
* **Straightforward for All Users:** With an intuitive interface requiring no specialized expertise, the Code Encrypter is accessible to programmers at any skill level. The simple and easy-to-use design allows anyone to quickly encrypt their code.

### Features

1. **File Selection**: Allows users to select a Python file for encryption.
2. **Code Viewing**: Users can view both the original and encrypted versions of the code.
3. **Automatic Encryption**: Encrypts the selected Python code automatically upon file selection.
4. **File Saving**: Provides an option to save the encrypted code to a file.
5. **About Information**: Displays information about the application and its developer.

### Usage Instructions

1. **Launch the Application**: Start the application to view the main interface.
2. **Select a File**: Click on the 'Choose File' button to select the desired Python file (.py) that you wish to encrypt.
3. **View Original Code**: Once a file is selected, you can click on the 'Original Code' button to view the code in its original form.
4. **Automatic Encryption**: Upon file selection, the code is automatically encrypted. The encrypted code is functional and can be executed, but its content is unreadable.
5. **View Encrypted Code**: To view the encrypted version of the code, click on the 'Emoji encrypted Code' or 'Var encrypted code' button to view the Encrypted code in required encryption type.
6. **Save Encrypted Code**: If you wish to save the encrypted code to a file, click on the 'Save File' button and choose a destination on your computer.
7. **About Button**: Click on the 'About' button to view information about the application and its developer.

### Example

Let's take a simple 'Hello World!' code as an example:

print('Hello World!')

When encrypted using the Code Encrypter, it transforms into:

exec("".join(map(chr,[int("".join(str({'🐅': 1, '👀': 9, '💎': 6, '💡': 2, '💰': 5, '😂': 4, '😏': 8, '🚫': 7, '🦻': 3, '\U0001fae5': 0}[i]) for i in x.split())) for x in

"🐅 🐅 💡 🐅 🐅 😂 🐅 🫥 💰 🐅 🐅 🫥 🐅 🐅 💎 😂 🫥 🦻 👀 🚫 💡 🐅 🫥 🐅 🐅 🫥 😏 🐅 🫥 😏 \

🐅 🐅 🐅 🦻 💡 😏 🚫 🐅 🐅 🐅 🐅 🐅 😂 🐅 🫥 😏 🐅 🫥 🫥 🦻 🦻 🦻 👀 😂 🐅".split(" ")])))

And when Encrypted using the Var encoding(which encodes using Base64) for the same code, it transforms into:

import base64; exec(base64.b64decode("cHJpbnQoJ0hlbGxvIFdvcmxkIScp".encode("ascii")).decode("ascii"))

These encrypted codes, when executed, will still output 'Hello World!' without requiring any decryption.

### Conclusion

The Code Encrypter is a robust tool for developers who prioritize the security and confidentiality of their code. By converting source code into an encrypted format, it ensures that the code's logic remains confidential, allowing for secure sharing and storage.

It offers a unique blend of security and functionality, ensuring that developers can protect their code without compromising its execution. Whether you're a hobbyist or a professional developer, the Code Encrypter is a valuable addition to your toolkit.