# Encryption and Decryption
## Miniproject Report

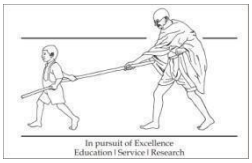
## Submitted by

1. Sakshi Chavan - 202001103061
2. Shreya Deshmane - 202001103066

To the Faculty of Engineering of
Mahatma Gandhi Mission University, Aurangabad

## Under Guidance of

Prof. Chaitali Patil




**Department of Computer Science & Engineering**
**MGM's Jawaharlal Nehru Engineering College, Aurangabad**
**Academic Year 2021-22 Part-II**

# ACKNOWLEDGEMENT

The completion of the project work is the milestone in student's life and its execution is in inevitable in the hand of guide. We would also like to express our deep regards and gratitude to our Principal Dr. H. H Shinde, Jawaharlal Nehru Engineering College, Aurangabad.

We also would take this opportunity to thank our HOD Dr. Vijaya Musande Jawaharlal Nehru Engineering College, Aurangabad, for her valuable guidance and appreciation for giving form and substance to this report.

I highly indebted and obliged to my project guide Prof. Chaitali Patil, Jawaharlal Nehru Engineering College, Aurangabad. Her aspiring guidance, constructive criticism and friendly advice helped us to complete this project study very effectively.

I am indebted to all the people who directly or indirectly helped me in the completion of this project.

I also want to take this opportunity to express my gratitude to my parents, friend and faculties who helped me directly or indirectly in completing this project successfully.

# INDEX

# ABSTRACT

**Objectives:**

- To protect data content, rather than preventing unauthorized interception of or access to data transmissions.
- For security organizations and in personal security software designed to protect user data.

**Website is created for:**

For securing sensitive data and to decrypt the encrypted the data

**Functionality:**

- Encryption prevent unauthorized parties from reading it
- Decryption converts an encrypted message back to its original format.
- Secures sensitive information.
- Cannot be Corrupted
- Enhanced Security

**Topics Included:**

1. Java programming.
2. Use of IDE
3. Use of DES(Data encryption standard)

# Introduction

This project is for security purposes, in this project you can encrypt your text file and again decrypt it. In this project, we have used the DES algorithm and cipher concept for encryption. The Encryption techniques hide the original content of a data in such a way that the original information is recovered only through using a key known as decryption process. The objective of the encryption is to secure or protect data from unauthorized access in term of viewing or modifying the data.

# Specifications

## Interpreter:

- Text Editor
- IDE

The Encryption techniques hide the original content of a data in such a way that the original information is recovered only through using a key known as decryption process. The objective of the encryption is to secure or protect data from unauthorized access in term of viewing or modifying the data.

# Implementation of program:

ED - Notepad

File Edit View

```
import java.io.File;
import java.io.FileInputStream;
import java.io.FileOutputStream;
import java.io.IOException;
import java.io.InputStream;
import java.io.OutputStream;
import java.security.InvalidKeyException;
/*import java.io.FileInputStream;
import java.io.FileOutputStream;
import java.io.IOException;
import java.io.InputStream;
import java.io.OutputStream;
import java.security.InvalidAlgorithmParameterException;
import java.security.InvalidKeyException;*/
import java.security.NoSuchAlgorithmException;
import java.security.SecureRandom;
import java.security.spec.InvalidKeySpecException;
import java.util.Scanner;

import javax.crypto.Cipher;
import javax.crypto.CipherInputStream;
import javax.crypto.CipherOutputStream;
import javax.crypto.NoSuchPaddingException;
import javax.crypto.SecretKey;
import javax.crypto.SecretKeyFactory;
import javax.crypto.spec.DESKeySpec;

public class ED
{
    public static void encryptDecrypt(String key, int cipherMode, File in, File out)
    throws InvalidKeyException, NoSuchAlgorithmException, InvalidKeySpecException, NoSuchPaddingException,
    IOException
```

ED - Notepad

File Edit View

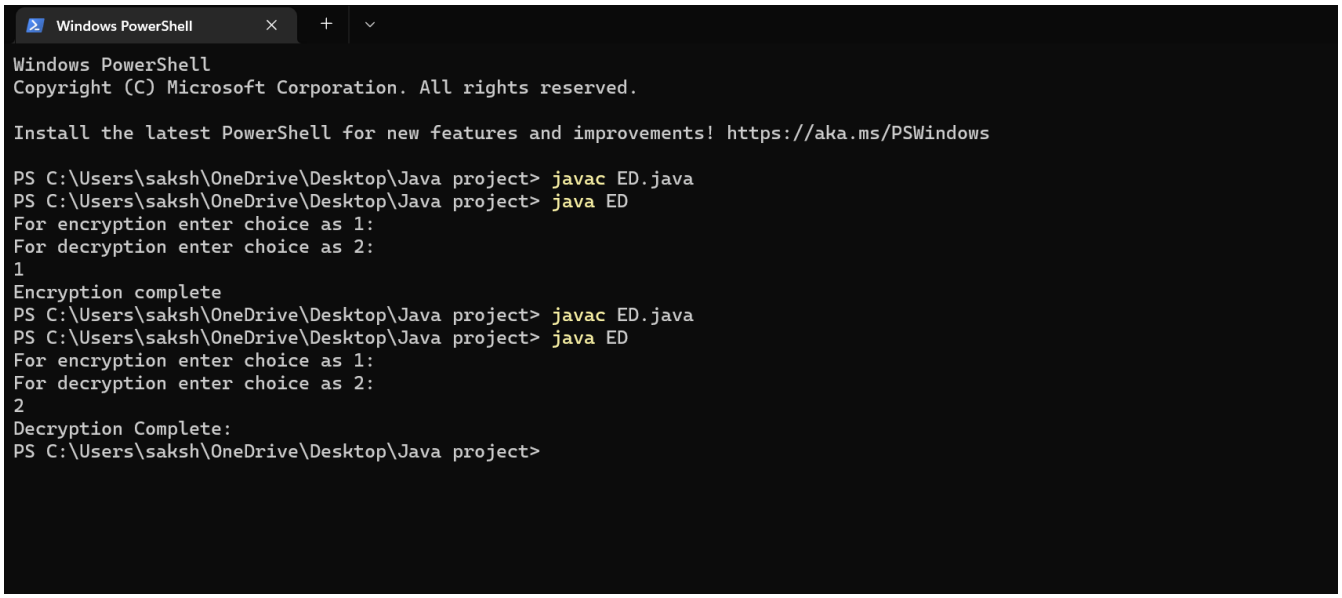
```
    {
        FileInputStream fis=new FileInputStream(in);
        FileOutputStream fos=new FileOutputStream(out);

        DESKeySpec desKeySpec=new DESKeySpec(key.getBytes());

        SecretKeyFactory skf= SecretKeyFactory.getInstance("DES");
        SecretKey secretKey=skf.generateSecret(desKeySpec);

        Cipher cipher=Cipher.getInstance("DES/ECB/PKCS5Padding");

        if(cipherMode==Cipher.ENCRYPT_MODE)
        {
            cipher.init(Cipher.ENCRYPT_MODE, secretKey, SecureRandom.getInstance("SHA1PRNG"));
            CipherInputStream cis=new CipherInputStream(fis,cipher);
            write(cis,fos);
        }
        else if(cipherMode==Cipher.DECRYPT_MODE)
        {
           cipher.init(Cipher.DECRYPT_MODE, secretKey, SecureRandom.getInstance("SHA1PRNG"));
           CipherOutputStream cos=new CipherOutputStream(fos, cipher);
           write(fis, cos);
        }

    }
    private static void write(InputStream in, OutputStream out)throws IOException
    {
        byte[] buffer=new byte[64];
        int numOfBytesRead;
        while((numOfBytesRead=in.read(buffer))!=-1)
        {
        out.write(buffer,0,numOfBytesRead);
    }
    out.close();
```

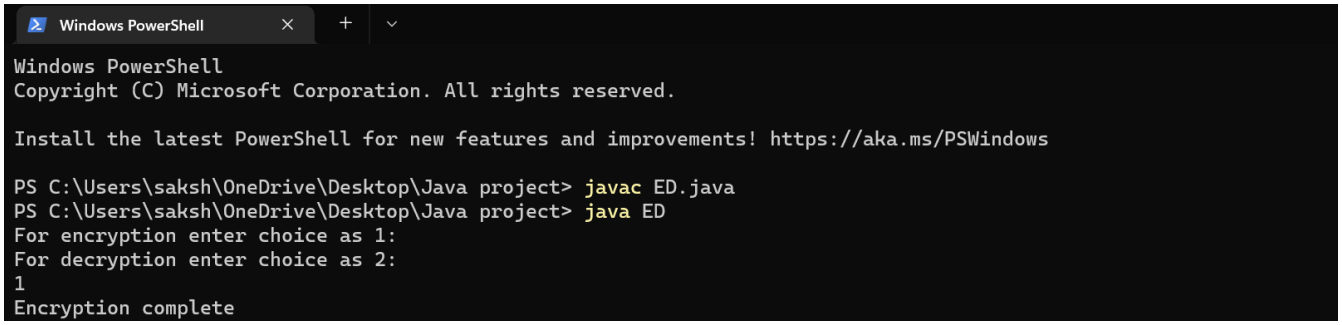
ED - Notepad

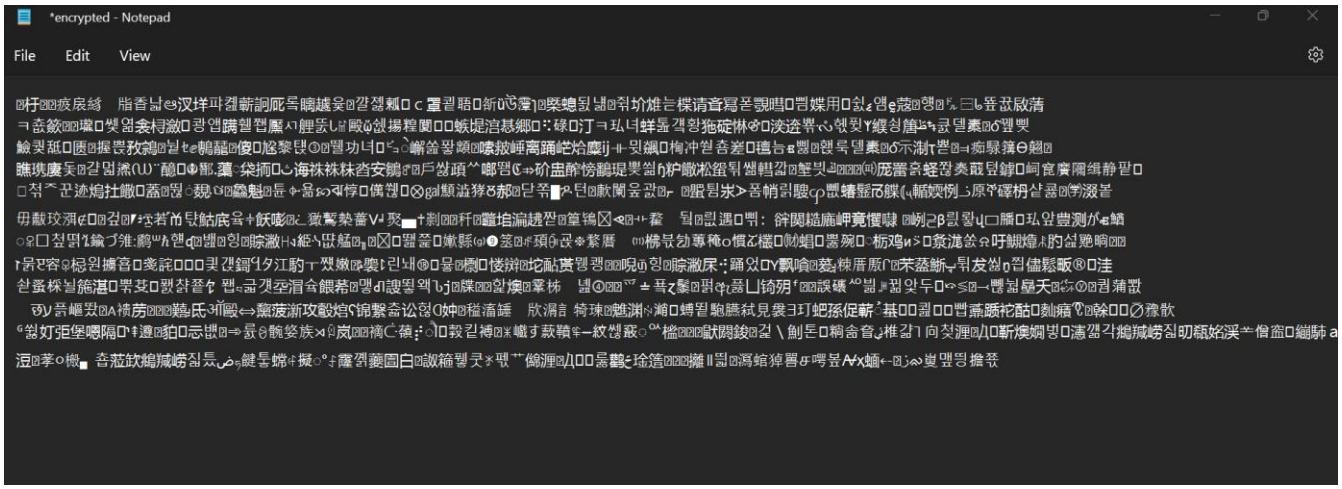
File Edit View

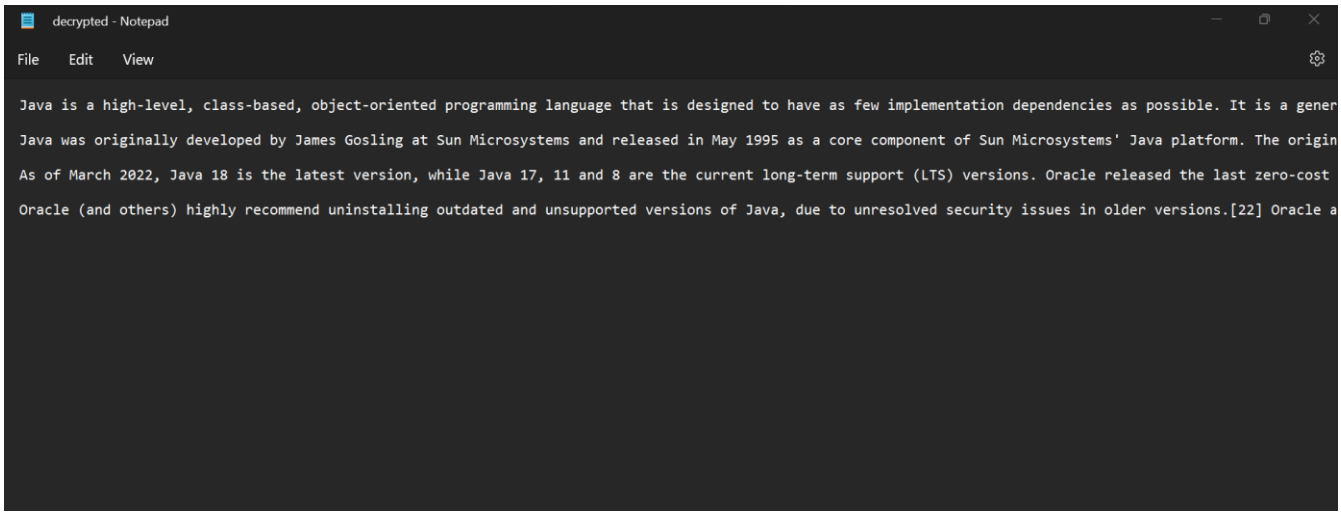
```
    in.close();
}
  public static void main(String[] args) {
        Scanner sc=new Scanner(System.in);
        System.out.println("For encryption enter choice as 1: ");
             System.out.println("For decryption enter choice as 2: ");
            int choice=sc.nextInt();
      File plaintext = new File("java.txt");  //file path which has plain text
      File encrypted = new File("encrypted.txt"); //blank file path which will contain encrypted text after encryption
      if(choice==1)
      {
    try {
        encryptDecrypt("12345678", Cipher.ENCRYPT_MODE, plaintext, encrypted);
        System.out.println("Encryption complete");

    } catch(Exception e) {
        e.printStackTrace();
    }
      }

if(choice==2)
{
File encrypted2=new File("encrypted.txt"); //encrypted file
File decrypted=new File("decrypted.txt"); //empty text file which will contain decrypted text after applying decryption

try
{
      encryptDecrypt("12345678",Cipher.DECRYPT_MODE,encrypted2,decrypted);
      System.out.println("Decryption Complete:");
}
catch(InvalidKeyException | NoSuchAlgorithmException | InvalidKeySpecException | NoSuchPaddingException | IOException e)
{
      e.printStackTrace();
}

}
  }
}
```

# Result


Windows PowerShell
Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.
Install the latest PowerShell for new features and improvements! https://aka.ms/PSWindows
PS C:\Users\saksh\OneDrive\Desktop\Java project> javac ED.java
PS C:\Users\saksh\OneDrive\Desktop\Java project> java ED
For encryption enter choice as 1:
For decryption enter choice as 2:
1
Encryption complete
PS C:\Users\saksh\OneDrive\Desktop\Java project> javac ED.java
PS C:\Users\saksh\OneDrive\Desktop\Java project> java ED
For encryption enter choice as 1:
For decryption enter choice as 2:
2
Decryption Complete:
PS C:\Users\saksh\OneDrive\Desktop\Java project>


Windows PowerShell
Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.
Install the latest PowerShell for new features and improvements! https://aka.ms/PSWindows
PS C:\Users\saksh\OneDrive\Desktop\Java project> javac ED.java
PS C:\Users\saksh\OneDrive\Desktop\Java project> java ED
For encryption enter choice as 1:
For decryption enter choice as 2:
1
Encryption complete


*encrypted - Notepad
File
Edit
View




```
PS C:\Users\saksh\OneDrive\Desktop\Java project> javac ED.java
PS C:\Users\saksh\OneDrive\Desktop\Java project> java ED
For encryption enter choice as 1:
For decryption enter choice as 2:
2
Decryption Complete:
PS C:\Users\saksh\OneDrive\Desktop\Java project>
```

# Conclusion

AES Java provides a wide range of modes and features for encryption of numerous types of data. It offers authenticity and integrity for encrypted data, Parallel encryption and option for making a cipher stream out of a cipher block. All these modes make it very convenient for java developers to encrypt the data as per their requirements and type of data.

## References:

https://www.baeldung.com/java-aes-encryption-decryption

https://howtodoinjava.com/java/java-security/java-aes-encryption-example/

https://www.section.io/engineering-education/implementing-aes-encryption-and-decryption-in-java/