

“Project Based Learning”

B.Tech.-Cse(A)

Session-2016-17

“Encryption / Decryption Cipher”

(Directed by-JAVA CORE PROGRAMMING LANGUAGE)

(Sub. Code-CSL0405)

Submitted to- Submitted by-

- Kapil Sharma -- Atul Anand (15028)

Bharti Parmar (15032)

Kamlesh Sharma (15048)

**“Encryption-Decryption cipher”**

(Dare to Decode…)

Special Thanks:

Aman Gautam

(B. Tech CSE-A)

**Table of Contents:**

**1. Objective**

**2. Introduction**

**3. Software Requirements**

**4. Team Members**

**5. Methodology**

**6. Code**

**7. Output (Screen Shots)**

**8. Cheat Sheet**

**9. Learning Outcome**

**10. Limitations of this project**

**11. References**

**Objective:**

The main objective of this project is to build a portable program for Encryption and Decryption using Packages.

Users can implement these packages into their program to run Cipher. Here, we have shown a glimpse of its implementation in a fully functional applet programming.

**Introduction:**

This project has made us understand the concepts and implementations of package in a better way. The key idea behind this project is to build a portable cipher for daily and general use. Most of the Ciphers available in industry are based on some Algorithms (RSA Algo- Most famous one). They somehow provide some strong and secure method of data encryption/decryption. But this Cipher built by us shows uniqueness in itself. We have built this Cipher without any algo or logic, which makes it impossible to decode by cyber attackers. One can decode it by using the Cheat Sheet only owned by the developer itself.

We have implemented this Cipher in an Applet. One can build a jar file out of it, and can implement this cipher in messaging and communication systems (Web Apps) to hide their texts and privacy.

**Team Members:**

**1. Name: Bharti Parmar**

**Roll: BETN1CS15032**

**Contribution:**

* **key idea behind the project**
* **Cipher main code**
* **Documentation**

**2. Name: Kamlesh Sharma**

**Roll: BETN1CS15048**

**Contribution:**

* **Applet Programming**
* **Error Checking**
* **Final Testing**

**3. Name: Atul Anand**

**Roll: BETN1CS15028**

**Contribution:**

* **Encryption/Decryption Logic**
* **Package building / testing for importing to programs.**
* **Documentation**

**Software Requirements:**

1. Java software kit (JDK/ JRE - Ver.\_1.8)

2. Text Editor (Notepad/ Ecllipse IDE)

3. MS-DOS

**Methodology:**

* There are two source files kept in the package serving different purposes; namely:

1. Cipher.java: containing parameterized methods encrypt() and decrypt(). Its class file is implemented in the Applet programming.
2. CipherImplement.java: containing non-parameterized methods encrypt() and decrypt(), both of which are taking inputs of String through Scanner() method.

* Structure of the package:

PBL\_TESTING\ -----Base Directory

Com->pbl->cipher -----Package name

* Two Test Files, namely:

1. expack.java: Implementing the CipherImplement.class file. Program is tested on MS-DOS after compiling and interpreting the source file.
2. CipherApplet.java: Implementing the Cipher.class file. Program is tested on an Applet. Two buttons are given to encrypt the message thrown in text box, and then another button to decrypt that encrypt message to receive the original message again.

* This program is capable to encrypt/ decrypt a single word or a complete sentence as per requirement.
* To run DOS Test file:

>> java Testcipher

* To run Applet program from DOS (Applet viewer):

>> appletviewer CipherApplet.java

**Code (Source Files):**

1. **Cipher.java**

**package com.pbl.cipher;**

**import java.util.Scanner;**

**public class Cipher {**

**String ins;**

**char[] outs;**

**char get, put, at, on;**

**//Scanner KB=new Scanner(System.in);**

**public void match(char at){**

**this.at=at;**

**if(get=='A'){**

**get='B';**

**}**

**else if(get=='B'){**

**get='N';**

**}**

**else if(get=='C'){**

**get='Z';**

**}**

**else if(get=='D'){**

**get='2';**

**}**

**else if(get=='E'){**

**get='K';**

**}**

**else if(get=='F'){**

**get='A';**

**}**

**else if(get=='G'){**

**get='9';**

**}**

**else if(get=='H'){**

**get='Q';**

**}**

**else if(get=='I'){**

**get='J';**

**}**

**else if(get=='J'){**

**get='I';**

**}**

**else if(get=='K'){**

**get='D';**

**}**

**else if(get=='L'){**

**get='O';**

**}**

**else if(get=='M'){**

**get='U';**

**}**

**else if(get=='N'){**

**get='1';**

**}**

**else if(get=='O'){**

**get='8';**

**}**

**else if(get=='P'){**

**get='C';**

**}**

**else if(get=='Q'){**

**get='H';**

**}**

**else if(get=='R'){**

**get='5';**

**}**

**else if(get=='S'){**

**get='7';**

**}**

**else if(get=='T'){**

**get='F';**

**}**

**else if(get=='U'){**

**get='M';**

**}**

**else if(get=='V'){**

**get='0';**

**}**

**else if(get=='W'){**

**get='6';**

**}**

**else if(get=='X'){**

**get='E';**

**}**

**else if(get=='Y'){**

**get='P';**

**}**

**else if(get=='Z'){**

**get='T';**

**}**

**else if(get=='0'){**

**get='V';**

**}**

**else if(get=='1'){**

**get='G';**

**}**

**else if(get=='2'){**

**get='4';**

**}**

**else if(get=='3'){**

**get='L';**

**}**

**else if(get=='4'){**

**get='S';**

**}**

**else if(get=='5'){**

**get='W';**

**}**

**else if(get=='6'){**

**get='3';**

**}**

**else if(get=='7'){**

**get='Y';**

**}**

**else if(get=='8'){**

**get='R';**

**}**

**else if(get=='9'){**

**get='X';**

**}**

**}**

**public void scratch(char on){**

**this.on=on;**

**if(put=='A'){**

**put='F';**

**}**

**else if(put=='B'){**

**put='A';**

**}**

**else if(put=='C'){**

**put='P';**

**}**

**else if(put=='D'){**

**put='K';**

**}**

**else if(put=='E'){**

**put='X';**

**}**

**else if(put=='F'){**

**put='T';**

**}**

**else if(put=='G'){**

**put='1';**

**}**

**else if(put=='H'){**

**put='Q';**

**}**

**else if(put=='I'){**

**put='J';**

**}**

**else if(put=='J'){**

**put='I';**

**}**

**else if(put=='K'){**

**put='E';**

**}**

**else if(put=='L'){**

**put='3';**

**}**

**else if(put=='M'){**

**put='U';**

**}**

**else if(put=='N'){**

**put='B';**

**}**

**else if(put=='O'){**

**put='L';**

**}**

**else if(put=='P'){**

**put='Y';**

**}**

**else if(put=='Q'){**

**put='H';**

**}**

**else if(put=='R'){**

**put='8';**

**}**

**else if(put=='S'){**

**put='4';**

**}**

**else if(put=='T'){**

**put='Z';**

**}**

**else if(put=='U'){**

**put='M';**

**}**

**else if(put=='V'){**

**put='0';**

**}**

**else if(put=='W'){**

**put='5';**

**}**

**else if(put=='X'){**

**put='9';**

**}**

**else if(put=='Y'){**

**put='7';**

**}**

**else if(put=='Z'){**

**put='C';**

**}**

**else if(put=='0'){**

**put='V';**

**}**

**else if(put=='1'){**

**put='N';**

**}**

**else if(put=='2'){**

**put='D';**

**}**

**else if(put=='3'){**

**put='6';**

**}**

**else if(put=='4'){**

**put='2';**

**}**

**else if(put=='5'){**

**put='R';**

**}**

**else if(put=='6'){**

**put='W';**

**}**

**else if(put=='7'){**

**put='S';**

**}**

**else if(put=='8'){**

**put='O';**

**}**

**else if(put=='9'){**

**put='G';**

**}**

**}**

**public String encrypt(String s){**

**//System.out.println("\n\t\t\t enter ur input:");**

**//ins= KB.next();**

**s=s.toUpperCase();**

**outs=new char[s.length()];**

**for(int i=0; i<s.length();i++){**

**get=s.charAt(i);**

**match(get);**

**outs[i]=get;**

**}**

**//System.out.print("\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*!!! ");**

**//System.out.print(outs);**

**//System.out.println(" !!!\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*");**

**String str=String.valueOf(outs);**

**return str;**

**}**

**public String decrypt(String s){**

**//System.out.println("\n\n\n\t\t\t enter ur input:\t\t\t");**

**//ins= KB.next();**

**outs=new char[s.length()];**

**for(int i=0; i<s.length();i++){**

**put=s.charAt(i);**

**scratch(put);**

**outs[i]=put;**

**}**

**//System.out.print("\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*!!! ");**

**//System.out.print(outs);**

**//System.out.println(" !!!\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*");**

**String str=new String(outs);**

**return str;**

**}**

**}**

**================#############==================**

1. **CipherImplement.java**

**package com.pbl.cipher;**

**import java.util.Scanner;**

**public class CipherImplement {**

**String ins;**

**char[] outs;**

**char get, put, at, on;**

**Scanner KB=new Scanner(System.in);**

**public void match(char at){**

**this.at=at;**

**if(get=='A'){**

**get='B';**

**}**

**else if(get=='B'){**

**get='N';**

**}**

**else if(get=='C'){**

**get='Z';**

**}**

**else if(get=='D'){**

**get='2';**

**}**

**else if(get=='E'){**

**get='K';**

**}**

**else if(get=='F'){**

**get='A';**

**}**

**else if(get=='G'){**

**get='9';**

**}**

**else if(get=='H'){**

**get='Q';**

**}**

**else if(get=='I'){**

**get='J';**

**}**

**else if(get=='J'){**

**get='I';**

**}**

**else if(get=='K'){**

**get='D';**

**}**

**else if(get=='L'){**

**get='O';**

**}**

**else if(get=='M'){**

**get='U';**

**}**

**else if(get=='N'){**

**get='1';**

**}**

**else if(get=='O'){**

**get='8';**

**}**

**else if(get=='P'){**

**get='C';**

**}**

**else if(get=='Q'){**

**get='H';**

**}**

**else if(get=='R'){**

**get='5';**

**}**

**else if(get=='S'){**

**get='7';**

**}**

**else if(get=='T'){**

**get='F';**

**}**

**else if(get=='U'){**

**get='M';**

**}**

**else if(get=='V'){**

**get='0';**

**}**

**else if(get=='W'){**

**get='6';**

**}**

**else if(get=='X'){**

**get='E';**

**}**

**else if(get=='Y'){**

**get='P';**

**}**

**else if(get=='Z'){**

**get='T';**

**}**

**else if(get=='0'){**

**get='V';**

**}**

**else if(get=='1'){**

**get='G';**

**}**

**else if(get=='2'){**

**get='4';**

**}**

**else if(get=='3'){**

**get='L';**

**}**

**else if(get=='4'){**

**get='S';**

**}**

**else if(get=='5'){**

**get='W';**

**}**

**else if(get=='6'){**

**get='3';**

**}**

**else if(get=='7'){**

**get='Y';**

**}**

**else if(get=='8'){**

**get='R';**

**}**

**else if(get=='9'){**

**get='X';**

**}**

**}**

**public void scratch(char on){**

**this.on=on;**

**if(put=='A'){**

**put='F';**

**}**

**else if(put=='B'){**

**put='A';**

**}**

**else if(put=='C'){**

**put='P';**

**}**

**else if(put=='D'){**

**put='K';**

**}**

**else if(put=='E'){**

**put='X';**

**}**

**else if(put=='F'){**

**put='T';**

**}**

**else if(put=='G'){**

**put='1';**

**}**

**else if(put=='H'){**

**put='Q';**

**}**

**else if(put=='I'){**

**put='J';**

**}**

**else if(put=='J'){**

**put='I';**

**}**

**else if(put=='K'){**

**put='E';**

**}**

**else if(put=='L'){**

**put='3';**

**}**

**else if(put=='M'){**

**put='U';**

**}**

**else if(put=='N'){**

**put='B';**

**}**

**else if(put=='O'){**

**put='L';**

**}**

**else if(put=='P'){**

**put='Y';**

**}**

**else if(put=='Q'){**

**put='H';**

**}**

**else if(put=='R'){**

**put='8';**

**}**

**else if(put=='S'){**

**put='4';**

**}**

**else if(put=='T'){**

**put='Z';**

**}**

**else if(put=='U'){**

**put='M';**

**}**

**else if(put=='V'){**

**put='0';**

**}**

**else if(put=='W'){**

**put='5';**

**}**

**else if(put=='X'){**

**put='9';**

**}**

**else if(put=='Y'){**

**put='7';**

**}**

**else if(put=='Z'){**

**put='C';**

**}**

**else if(put=='0'){**

**put='V';**

**}**

**else if(put=='1'){**

**put='N';**

**}**

**else if(put=='2'){**

**put='D';**

**}**

**else if(put=='3'){**

**put='6';**

**}**

**else if(put=='4'){**

**put='2';**

**}**

**else if(put=='5'){**

**put='R';**

**}**

**else if(put=='6'){**

**put='W';**

**}**

**else if(put=='7'){**

**put='S';**

**}**

**else if(put=='8'){**

**put='O';**

**}**

**else if(put=='9'){**

**put='G';**

**}**

**}**

**public void encrypt(){**

**System.out.println("\n\t\t\t enter ur input:");**

**ins= KB.next();**

**ins=ins.toUpperCase();**

**outs=new char[ins.length()];**

**for(int i=0; i<ins.length();i++){**

**get=ins.charAt(i);**

**match(get);**

**outs[i]=get;**

**}**

**System.out.print("\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*!!! ");**

**System.out.print(outs);**

**System.out.println(" !!!\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*");**

**//String str=String.valueOf(outs);**

**//return str;**

**}**

**public void decrypt(){**

**System.out.println("\n\n\n\t\t\t enter ur input:\t\t\t");**

**ins= KB.next();**

**ins=ins.toUpperCase();**

**outs=new char[ins.length()];**

**for(int i=0; i<ins.length();i++){**

**put=ins.charAt(i);**

**scratch(put);**

**outs[i]=put;**

**}**

**System.out.print("\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*!!! ");**

**System.out.print(outs);**

**System.out.println(" !!!\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*");**

**//String str=new String(outs);**

**//return str;**

**}}**

**==================###############================**

**# Test Files:**

1. **CipherApplet.java**

**import com.pbl.cipher.Cipher;**

**import java.applet.Applet;**

**import java.awt.\*;**

**import java.awt.event.\*;**

**/\*<applet code="CipherApplet.class" height=200 width=550>**

**</applet>**

**\*/**

**public class CipherApplet extends Applet implements ActionListener{**

**TextField t1;**

**TextField t2;**

**TextField t3;**

**Button e;**

**Button d;**

**Button c;**

**String s,e1,d1;**

**Cipher ob;**

**public void init()**

**{**

**t1=new TextField(20);**

**t2=new TextField(20);**

**t3=new TextField(20);**

**ob=new Cipher();**

**e=new Button("Encrypt");**

**e.setBackground(Color.green);**

**d=new Button("Decrypt");**

**d.setBackground(Color.red);**

**c=new Button(" Reset ");**

**c.setBackground(Color.blue);**

**add(t1);**

**add(t2);**

**add(t3);**

**add(e);**

**add(d);**

**add(c);**

**e.addActionListener(this);**

**d.addActionListener(this);**

**c.addActionListener(this);**

**}**

**public void actionPerformed(ActionEvent ae)**

**{**

**if(ae.getSource()==e)**

**{**

**s=String.valueOf(t1.getText());**

**e1=ob.encrypt(s);**

**t2.setText(e1);**

**}**

**if(ae.getSource()==d)**

**{**

**d1=ob.decrypt(t2.getText());**

**t3.setText(d1);**

**}**

**if(ae.getSource()==c)**

**{**

**t1.setText(" ");**

**t2.setText("");**

**t3.setText("");**

**}}}**

**==============##############==================**

1. **expack.java**

**import com.pbl.cipher.CipherImplement;**

**class Testcipher{**

**public static void main(String[] args){**

**CipherImplement ob=new CipherImplement();**

**System.out.println("\n YOU MUST TRY TO ENCRYPT THE CODE:\n");**

**ob.encrypt();**

**System.out.println("\n NOW I'M GONNA DECRYPT THIS CODE:\n");**

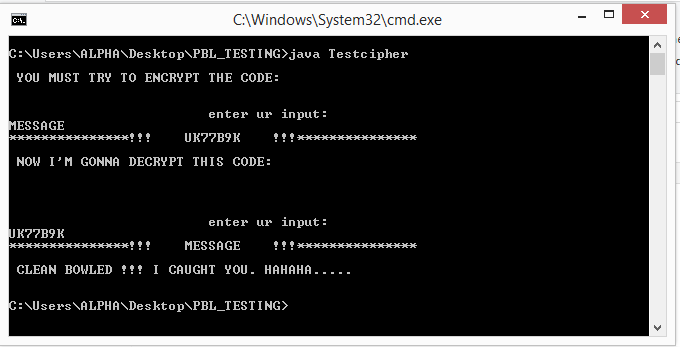
**ob.decrypt();**

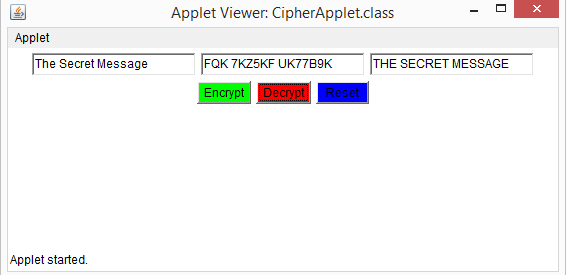
**System.out.println("\n CLEAN BOWLED !!! I CAUGHT YOU. HAHAHA.....\n");**

**}}**

**==================################================**

**Output (Screen Shots):**

****

****

**Cheat Sheet:**

|  |  |  |  |
| --- | --- | --- | --- |
| Encryption | | decryption | |
| **A** | **B** | **A** | **F** |
| **B** | **N** | **B** | **A** |
| **C** | **Z** | **C** | **P** |
| **D** | **2** | **D** | **K** |
| **E** | **K** | **E** | **X** |
| **F** | **A** | **F** | **T** |
| **G** | **9** | **G** | **1** |
| **H** | **Q** | **H** | **Q** |
| **I** | **J** | **I** | **J** |
| **J** | **I** | **J** | **I** |
| **K** | **D** | **K** | **E** |
| **L** | **O** | **L** | **3** |
| **M** | **U** | **M** | **U** |
| **N** | **1** | **N** | **B** |
| **O** | **8** | **O** | **L** |
| **P** | **C** | **P** | **Y** |
| **Q** | **H** | **Q** | **H** |
| **R** | **5** | **R** | **8** |
| **S** | **7** | **S** | **4** |
| **T** | **F** | **T** | **Z** |
| **U** | **M** | **U** | **M** |
| **V** | **0** | **V** | **0** |
| **W** | **6** | **W** | **5** |
| **X** | **E** | **X** | **9** |
| **Y** | **P** | **Y** | **7** |
| **Z** | **T** | **Z** | **C** |
| **0** | **V** | **0** | **V** |
| **1** | **G** | **1** | **N** |
| **2** | **4** | **2** | **D** |
| **3** | **L** | **3** | **6** |
| **4** | **S** | **4** | **2** |
| **5** | **W** | **5** | **R** |
| **6** | **3** | **6** | **W** |
| **7** | **Y** | **7** | **S** |
| **8** | **R** | **8** | **O** |
| **9** | **X** | **9** | **G** |

**Learning Outcome:**

Through this project based Learning methodology, we have actually got an opportunity to explore the subjects taught to us in our Academics. We have gained an experience to develop something totally functional out of raw academic stuffs.

This project can be further expanded to its next level. We can put into action for a chat messenger app, or any web application based on communication between two terminals. Additionally, we have gained the in depth concepts behind Cipher technologies.

**Limitations of this project:**

We have restricted this project to some limitations:

* The Dos implementation does work for a complete Sentence input.
* Any further development in this project is impossible without developer’s presence.
* Lower Case letters give wrong output in DOS implementation. ( Case conversation is not applied here.). Though, it works well in Applet Implementation.

**References:**

1. [**http://www.javatpoint.com/package**](http://www.javatpoint.com/package)
2. [**http://www.dreamincode.net/forums/topic/85570-positioning-the-buttons-in-an-applet/**](http://www.dreamincode.net/forums/topic/85570-positioning-the-buttons-in-an-applet/)
3. [**http://stackoverflow.com/**](http://stackoverflow.com/)
4. **https://www.tutorialspoint.com/awt/awt\_event\_handling.htm**