**Ceaser cipher-encryption**

import java.util.\*;

public class **CaesarCipherProgram** {

public static void main(String args[]) {

Scanner sc = new Scanner(System.in);

System.out.println(" Input the plaintext message : ");

String plaintext = sc.nextLine();

System.out.println(" Enter the value by which each character in the plaintext message gets shifted : ");

int shift = sc.nextInt();

String ciphertext = "";

char alphabet;

for(int i=0; i < plaintext.length();i++)

{

// Shift one character at a time

alphabet = plaintext.charAt(i);

// if alphabet lies between a and z

if(alphabet >= 'a'||alphabet >='A' && alphabet <= 'z' || alphabet <='Z')

{

// shift alphabet

alphabet = (char) (alphabet + shift);

}

ciphertext = ciphertext + alphabet;

}

System.out.println(" ciphertext : " + ciphertext);

}

}  
  
**Decryption**

import java.util.\*;

public class Main {

public static void main(String args[]) {

Scanner sc = new Scanner(System.in);

System.out.println(" Input the plaintext message : ");

String plaintext = sc.nextLine();

System.out.println(" Enter the value by which each character in the plaintext message gets shifted : ");

int shift = sc.nextInt();

String ciphertext = "";

char alphabet;

for(int i=0; i < plaintext.length();i++)

{

// Shift one character at a time

alphabet = plaintext.charAt(i);

// if alphabet lies between a and z

if(alphabet >= 'a'||alphabet >='A' && alphabet <= 'z' || alphabet <='Z')

{

// shift alphabet

alphabet = (char) (alphabet - shift);

}

ciphertext = ciphertext + alphabet;

}

System.out.println(" ciphertext : " + ciphertext);

}

}

Substitution cipher

https://www.geeksforgeeks.org/java-program-to-implement-the-monoalphabetic-cypher/

**Approach:**

**1.**  Create two char arrays, one for normal alphabets(say normalChar[]) and another is for encoding(say codedChar[]).

**2.**  We will use two functions:

* **stringEncryption:** We pass string(string with all characters in lower case) as a parameter. Initialize an empty string(say encryptedString**)**. We use for loop and compare each character with normal char array, whenever the condition is true, add the character with the corresponding index of codedChar to the encrypted string. In the case of special characters, we will add them directly to the string.
* **stringDescryption:** We pass the encrypted string as the parameter. Initialize an empty string(say descryptedString). In the same way, we run the for loop and add the character with the corresponding index of normalChar to the decrypted string. In the case of special characters, we will add them directly to the string.

// Java Program to Implement the Monoalphabetic Cypher

import java.io.\*;

class GFG {

public static char normalChar[]

= { 'a', 'b', 'c', 'd', 'e', 'f', 'g', 'h', 'i',

'j', 'k', 'l', 'm', 'n', 'o', 'p', 'q', 'r',

's', 't', 'u', 'v', 'w', 'x', 'y', 'z' };

public static char codedChar[]

= { 'Q', 'W', 'E', 'R', 'T', 'Y', 'U', 'I', 'O',

'P', 'A', 'S', 'D', 'F', 'G', 'H', 'J', 'K',

'L', 'Z', 'X', 'C', 'V', 'B', 'N', 'M' };

// Function which returns encrypted string

public static String stringEncryption(String s)

{

// initializing an empty String

String encryptedString = "";

// comparing each character of the string and

// encoding each character using the indices

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

for (int j = 0; j < 26; j++) {

// comparing the character and

// adding the corresponding char

// to the encryptedString

if (s.charAt(i) == normalChar[j])

{

encryptedString += codedChar[j];

break;

}

// if there are any special characters

// add them directly to the string

if (s.charAt(i) < 'a' || s.charAt(i) > 'z')

{

encryptedString += s.charAt(i);

break;

}

}

}

// return encryptedString

return encryptedString;

}

// Function which returns descryptedString

public static String stringDecryption(String s)

{

// Initializing the string

String decryptedString = "";

// Run the for loop for total string

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

{

for (int j = 0; j < 26; j++) {

// compare each characters and decode them

// using indices

if (s.charAt(i) == codedChar[j])

{

decryptedString += normalChar[j];

break;

}

// Add the special characters directly

if (s.charAt(i) < 'A' || s.charAt(i) > 'Z')

{

decryptedString += s.charAt(i);

break;

}

}

}

return decryptedString;

}

public static void main(String args[])

{

String str = "Welcome to geeksforgeeks";

System.out.println("Plain text: " + str);

String encryptedString = stringEncryption(str.toLowerCase());

// printing encryptedString

System.out.println("Encrypted message: "+ encryptedString);

// the decryptedString

System.out.println("Decrypted message: " + stringDecryption (encryptedString));

}

}

Plain text: !Welcome!

Encrypted message: !VTSEGDT!

Decrypted message: !welcome!