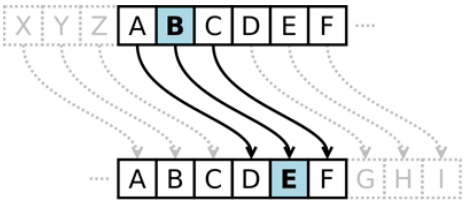
Kryptographie in Java

erste Standardverfahren

Caesar




```
import java.util.Scanner;

public class Caesar {
  public static void main (String[] args) {
    System.out.print("Bitte Verschiebung eingeben: ");
    Scanner scan = new Scanner(System.in); int shift = scan.nextInt();
    System.out.print("Bitte Text eingeben: ");
    Scanner scan2 = new Scanner(System.in); String a = scan2.nextLine();
    char[] text = a.toUpperCase().toCharArray();
    crypt(text, shift); System.out.println("Verschlüsselt: " + new String(text));
    System.out.println("entschlüsseln ? j/n");
    Scanner scan3 = new Scanner(System.in); String b = scan3.nextLine();
    if (b.equals("j")) { crypt(text, -shift); System.out.println("Entschlüsselt: " + new String(text)); }
  }

private static void crypt(char[] text, int shift) {
    for (int i = 0; i < text.length; i++) {
      if (text[i] >= 65 && text[i] <= 90) {
        text[i] = (char)(text[i] + shift);
        if(text[i] > 90) text[i] -= 26;
        else if(text[i] < 65) text[i] += 26;
} } } }
```

```
Bitte Verschiebung eingeben: 4
Bitte Text eingeben: Check this out!
Verschlüsselt: GLIGO XLMW SYX!
entschlüsseln ? j/n
j
Entschlüsselt: CHECK THIS OUT!
```

Vigenère

```
import java.util.Scanner;

public class Vigenere {
    public static void main(String[] args) {
        Scanner ent = new Scanner(System.in); Scanner key = new Scanner(System.in);
        Scanner word = new Scanner(System.in);
        //Eingabe
        System.out.println("Wort:"); String wort = word.nextLine().toUpperCase(); //nur Großbuchstaben
        System.out.println("Schlüssel:"); String schlüssel = key.nextLine().toUpperCase(); //alles groß
        int a = wort.length() ;
        int b = a / schlüssel.length(); // Schlüsselwort mindestens gleich lang wie zu verschlüsselndes Wort
        String hilfe = schlüssel; for (int i = 1; i <= b ; i++) schlüssel = schlüssel + hilfe;
        //verschlüssle, achte auf Alphabetsgrenzen (nach Z bei A weiter!)
        String verwort = ""; int n;
        for (int i = 0; i <= a-1 ; i++) {
            n = (int)wort.charAt(i) + ((int)schlüssel.charAt(i) - 65);
            if (n > 90) n = n - 26; verwort = verwort + (char)n;
```

```
    }
    //Ausgabe
    System.out.println("Verschlüsseltes Wort:");
    for (int i = 0; i <= a-1 ; i++ ) System.out.print("" + verwort.charAt(i));
    System.out.println(""); System.out.println("");
    System.out.println("*****************************"); System.out.println("");
    //Eingabe
    System.out.println("Bitte Schlüssel eingeben:");
    String entschlüssel = ent.nextLine().toUpperCase();
    b = a / entschlüssel.length(); //Entschlüsselwort mindestens gleich lang wie zu verschlüsselndes
    hilfe = entschlüssel; for (int i = 1; i <= b ; i++) entschlüssel = entschlüssel + hilfe;
    //entschlüssle
    String ursprung = "";
    for (int i = 0; i <= a-1 ; i++) {
       n = (int)verwort.charAt(i) - ((int)entschlüssel.charAt(i) - 65);
       if (n < 65) n = n + 26; ursprung = ursprung + (char)n;
    }
    System.out.println("Ursprüngliches Wort: ");
    for (int i = 0; i <= a-1; i++) System.out.print("" + ursprung.charAt(i));
    System.out.println(""); System.out.println("");
    System.out.println("*****************************"); System.out.println("");
  }
```

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*
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	A
C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B
D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C
E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D
F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E
G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F
H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G
I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H
J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I
K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J
L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K
M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L
N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M
O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N
P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O
Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P
R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q
S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R
T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S
U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T
V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U
W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V
X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W
Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X
Z	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

```
Wort:
Frieden
Schlüssel:
abba
Verschlüsseltes Wort:
FSJEDFO

*****************************

Bitte Schlüssel eingeben:
abba
Ursprüngliches Wort:
FRIEDEN

*****************************
```

Transposition

```
public class Transposition {
   public static void main(String[] args){
      String input = "paulpanzeristsuperlustig";
      int length = 5;
      String encrypted, decrypted;
      encrypted = encrypt(input, length); System.out.println(encrypted);
      decrypted = decrypt(encrypted, length); System.out.println(decrypted);
   }

   private static String encrypt(String text, int length){
      StringBuilder builder = new StringBuilder();
      for (int i = 1; i <= length; i++)
         for (int j = i - 1; j < text.length(); j += length)
            builder.append(text.charAt(j));
      return builder.toString();
   }

   private static String decrypt(String text, int length){
      char[] arr = new char[text.length()];
      for (int i = 1, x = 0; i <= length; i++) {
         for (int j = i - 1; j < text.length(); j += length) {
            arr[j] = text.charAt(x);
            x++;
         }
      }
      return new String(arr);
   }
}
```

```
paipsansetuztrileslgpruu
paulpanzeristsuperlustig
```

P	A	U	L	P
A	N	Z	E	R
I	S	T	S	U
P	E	R	L	U
S	T	I	G	

---

andere Varianten:

T	U	E	E	Z
R	N	I	N	I
E	K	M	Z	G
F	T	O	W	U
F	D	R	A	H
P	R	G	N	R

P	R	G	N	R
F	D	R	A	H
F	T	O	W	U
E	K	M	Z	G
R	N	I	N	I
T	U	E	E	Z

KRYPTOGRAFIE.DE

T	R	E	N	Z
R	D	I	A	I
E	T	M	W	G
F	K	O	Z	U
F	N	R	N	H
P	U	G	E	R