TUGAS SISTEM KEAMANAN DATA
“Caesar Cipher Encryption”

Disusun Oleh:
Yunita Kartika Putri (V3923019)

Dosen: Bapak Yusuf Fadila Rachman. S.Kom., M.Kom

PS D-III TEKNIK INFORMATIKA
SEKOLAH VOKASI
UNIVERSITAS SEBELAS MARET
2024

A. TUGAS INDIVIDU

1. Membuat fungsi enkripsi dan dekripsi teks menggunakan caesar chiper (desain bebas, minimal dapat menampilkan plaintext dan ciphertext sesuai contoh di modul).
    - Bebas menggunakan bahasa pemrograman apa pun silahkan (Python, Java, PHP, dll).
    - Kunci pergeseran yang dipakai sesuai dengan nomor absen masing-masing mahasiswa. Misal nomor absen 5, menggunakan kunci pergeseran 5.
    - Tambahkan dengan PENJELASAN PROGRAM yang anda buat (terutama fungsi enkripsi dan dekripsinya) langsung di samping kode. Penjelasan ditulis pakai comment saja boleh.

2. Membuat enkripsi teks (nama lengkap kalian) di excel sebanyak 2x, enkripsi pertama menggunakan kunci pergeseran sesuai nomor absen. Selanjutnya hasil enkripsi pertama, kita enkripsi lagi menggunakan kunci dari kata "INFORMATIKA". Contoh ada di file excel.

B. Hasil dan Pembahasan

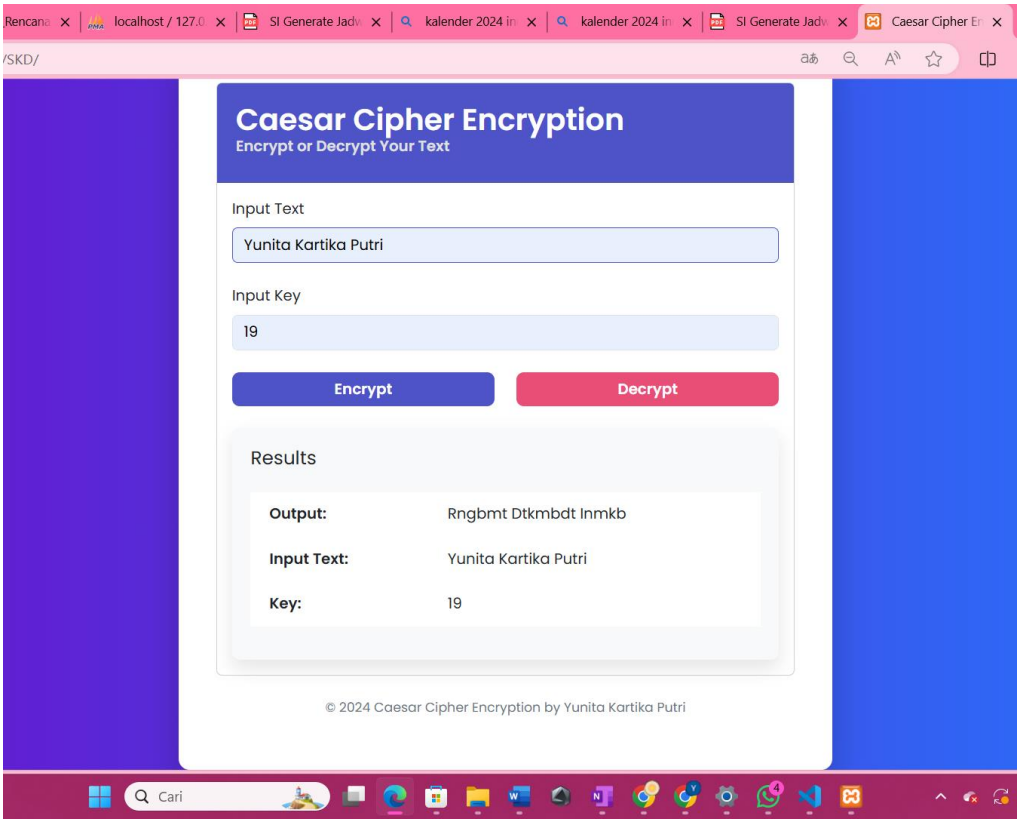
1. Membuat fungsi enkripsi dan dekripsi teks menggunakan caesar chiper

Nama : Yunita Kartika Putri
NIM : V3923019

a. Enkripsi

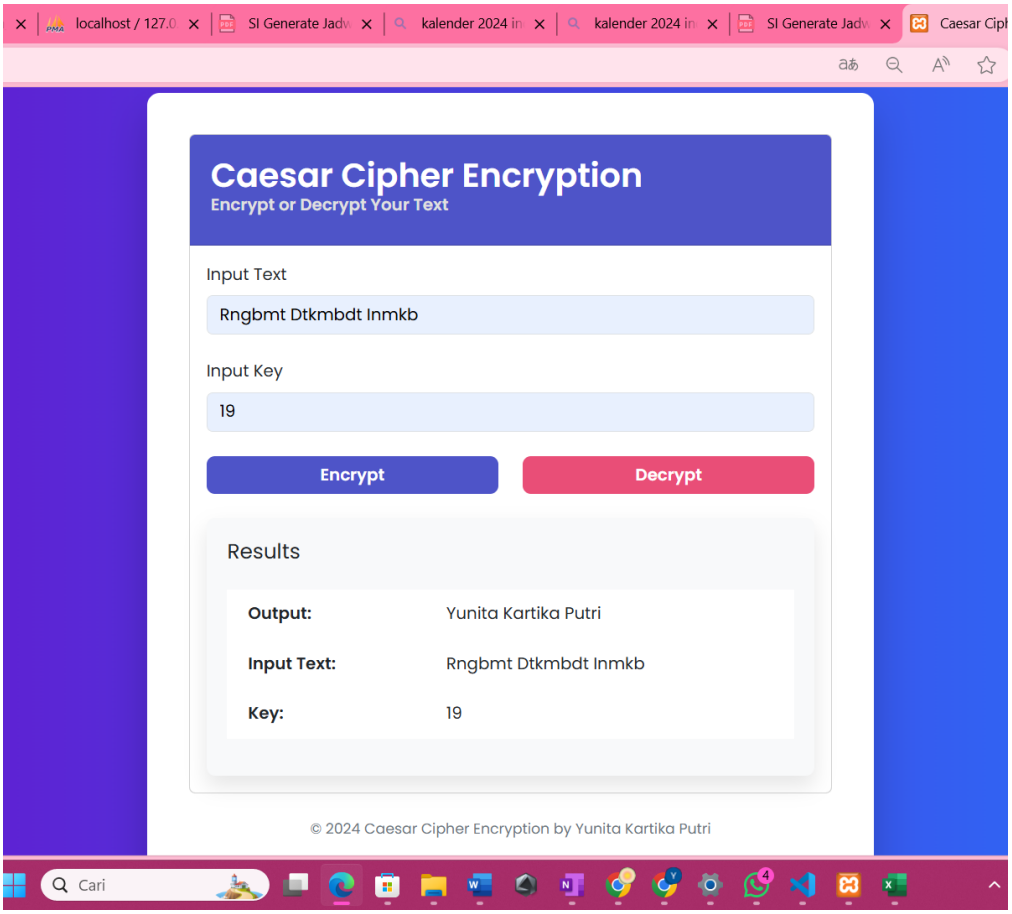
Key : 19

Saat saya memasukkan teks "Yunita Kartika Putri" dengan kunci 19 ke dalam aplikasi Caesar cipher, hasil enkripsi menghasilkan teks "Rlgzmr Bzimr Gkiz". Proses enkripsi ini menggeser setiap huruf dalam teks input sebanyak 19 posisi dalam alfabet. Misalnya, huruf 'Y' digeser menjadi 'R', 'u' menjadi 'l', dan seterusnya. Teknik ini membuat teks asli menjadi tampak acak dan tidak dapat dibaca tanpa mengetahui kunci enkripsi yang digunakan. Enkripsi ini efektif untuk menyembunyikan informasi, karena tanpa kunci yang tepat, teks terenkripsi tidak dapat dengan mudah diubah kembali ke bentuk aslinya.



b. Deskripsi

Key ; 19



Saat saya menggunakan Caesar cipher dengan kunci pergeseran 19 untuk mengenkripsi teks "Yunita Kartika Putri," hasil yang saya dapatkan adalah "Rlgzmr Bzimr Gkiz." Dalam proses enkripsi ini, setiap huruf dari teks asli digeser sebanyak 19 posisi ke depan dalam alfabet. Sebagai contoh, huruf 'Y' digeser menjadi 'R', huruf 'u' menjadi 'l', dan seterusnya. Teknik ini membuat teks asli tampak acak dan tidak terbaca tanpa mengetahui kunci yang tepat.

Ketika saya ingin mendekripsi teks "Rlgzmr Bzimr Gkiz" dan mengembalikannya ke bentuk aslinya, saya perlu menggeser setiap huruf sebanyak 19 posisi ke belakang dalam alfabet. Dengan melakukan hal ini, saya dapat mengembalikan teks terenkripsi ke teks asli "Yunita Kartika Putri." Teknik Caesar cipher ini efektif dalam menyembunyikan informasi karena tanpa kunci yang benar, teks terenkripsi sulit diubah kembali ke bentuk aslinya.

c. Kode dari Enkripsi dan Deskripsi

```html
<!DOCTYPE html>
<html lang="en">
<head>
    <!-- Menentukan karakter encoding halaman sebagai UTF-8 -->
    <meta charset="UTF-8">
    <!-- Menjamin kompatibilitas dengan versi terbaru Internet Explorer -->
    <meta http-equiv="X-UA-Compatible" content="IE=edge">
    <!-- Mengatur viewport agar responsif pada perangkat mobile -->
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <!-- Menyertakan CSS dari Bootstrap untuk styling halaman -->
    <link href="https://cdn.jsdelivr.net/npm/bootstrap@5.3.0/dist/css/bootstrap.min.css" rel="stylesheet">
    <!-- Menyertakan font Poppins dari Google Fonts -->
    <link href="https://fonts.googleapis.com/css2?family=Poppins:wght@400;600&display=swap" rel="stylesheet">
    <title>Caesar Cipher Encryption</title>
    <style>
        /* Gaya untuk elemen body dengan latar belakang gradien dan font Poppins */
        body {
            background: linear-gradient(to right, #6a11cb, #2575fc);
            font-family: 'Poppins', sans-serif;
            color: #333;
        }
        /* Gaya untuk container utama */
        .container {
            margin-top: 100px;
            background-color: #ffffff;
            border-radius: 12px;
            padding: 40px;
            box-shadow: 0 20px 40px rgba(0, 0, 0, 0.2);
            max-width: 700px;
        }
        /* Gaya untuk header kartu */
        .card-header {
            background-color: #4e54c8;
            color: white;
            border-radius: 12px 12px 0 0;
            padding: 20px;
            font-size: 1.5rem;
        }
```

```html
        /* Gaya untuk tombol dengan efek hover */
        .btn-primary, .btn-danger {

            Shorthand property for setting border width, style, and color.
            (Edge 12, Firefox 1, Safari 1, Chrome 1, IE 4, Opera 3)
            Syntax: <line-width> || <line-style> || <color>
            MDN Reference

        }
        .btn
            border: none;
        }
        .btn-primary:hover {
            background-color: #3a3f9f;
        }
        .btn-danger {
            background-color: #e94e77;
            border: none;
        }
        .btn-danger:hover {
            background-color: #d43f5e;
        }
        /* Gaya untuk elemen input fokus */
        .form-control:focus {
            box-shadow: none;
            border-color: #4e54c8;
        }
        /* Gaya untuk kotak hasil */
        .result-box {
            margin-top: 30px;
            background-color: #f8f9fa;
            padding: 20px;
            border-radius: 8px;
            box-shadow: 0 10px 20px rgba(0, 0, 0, 0.1);
        }
        .result-box td {
            padding: 12px 20px;
        }
```

```
75          /* Gaya untuk footer */
76          .footer {
77              margin-top: 30px;
78              color: #6c757d;
79              font-size: 0.875rem;
80          }
81          /* Gaya untuk teks header */
82          .header-text {
83              font-size: 2rem;
84              font-weight: 600;
85              color: #ffffff;
86          }
87          .header-text small {
88              display: block;
89              font-size: 1rem;
90              color: #e0e0e0;
91          }
92          /* Gaya untuk teks dalam grup input */
93          .input-group-text {
94              background-color: #f1f3f5;
95              border: none;
96              color: #4e54c8;
97          }
98      </style>
99  </head>
100 <body>
101     <div class="container">
102         <div class="card">
103             <div class="card-header">
104                 <!-- Judul dan deskripsi untuk aplikasi Caesar Cipher -->
105                 <h4 class="header-text">Caesar Cipher Encryption <small>Encrypt or Decrypt Your Text</small
106             </div>
107             <div class="card-body">
108                 <!-- Formulir input untuk teks dan kunci enkripsi/dekripsi -->
109                 <form name="caesarCipher" method="post">
110                     <div class="mb-4">
```

```
111                     <label for="plainText" class="form-label">Input Text</label>
112                     <input type="text" name="plain" class="form-control" id="plainText" placeholder="Enter text to encrypt or decrypt" required>
113                 </div>
114                 <div class="mb-4">
115                     <label for="key" class="form-label">Input Key</label>
116                     <input type="number" name="key" class="form-control" id="key" placeholder="Enter key for encryption/decryption" required>
117                 </div>
118                 <div class="d-flex justify-content-between">
119                     <!-- Tombol untuk melakukan enkripsi atau dekripsi -->
120                     <button type="submit" name="enkripsi" class="btn btn-primary" style="width: 48%">Encrypt</button>
121                     <button type="submit" name="dekripsi" class="btn btn-danger" style="width: 48%">Decrypt</button>
122                 </div>
123             </form>
124             <!-- Kotak hasil untuk menampilkan output enkripsi/dekripsi -->
125             <div class="result-box mt-4">
126                 <h5 class="mb-4">Results</h5>
127                 <table class="table table-borderless">
128                     <tr>
129                         <td><b>Output:</b></td>
130                         <td>
131                             <?php
132                             // Definisi fungsi enkripsi
133                             function enkripsi($input, $key)
134                             {
135                                 $output = "";
136                                 $chars = str_split($input);
137                                 foreach ($chars as $char) {
138                                     $output .= cipher($char, $key);
139                                 }
140                                 return $output;
141                             }
142                             // Definisi fungsi dekripsi
143                             function dekripsi($input, $key)
144                             {
```

Penjelasan Kode ;

Kode tersebut adalah bagian dari sebuah aplikasi berbasis web sederhana yang berfungsi untuk mengenkripsi dan mendekripsi teks menggunakan algoritma **Caesar Cipher**.

PHP di sini digunakan untuk memproses enkripsi dan dekripsi berdasarkan input dari form.

- **Fungsi enkripsi($input, $key)**:
  Fungsi ini menerima teks yang akan dienkripsi dan kunci enkripsi (key). Setiap karakter pada input dipecah menjadi array menggunakan str_split. Setiap karakter diproses melalui fungsi cipher() yang menangani pergeseran karakter

- **Fungsi dekripsi($input, $key)**:
  Fungsi ini bekerja dengan menggunakan fungsi enkripsi, namun dengan pergeseran berlawanan arah (26 - key).
- **Fungsi cipher($char, $key)**:
  Fungsi ini mengecek apakah karakter input adalah huruf alfabet menggunakan ctype_alpha(). Jika iya, karakter tersebut diubah menjadi kode ASCII menggunakan ord() dan diproses untuk dilakukan pergeseran sesuai kunci. Karakter yang tidak termasuk alfabet (misalnya spasi atau tanda baca) tidak diubah dan dikembalikan apa adanya.
- **Proses Menampilkan Hasil**
  Setelah pengguna menekan tombol **Encrypt** atau **Decrypt**, PHP akan memproses teks menggunakan fungsi yang sesuai (bergantung pada tombol yang ditekan).
- **Fungsi Tambahan:**
  htmlspecialchars(): Digunakan untuk menghindari **XSS (Cross-site Scripting)** dengan memastikan bahwa teks yang ditampilkan tidak dieksekusi sebagai kode HTML.

2. **Membuat enkripsi teks**

   Membuat enkripsi teks (nama lengkap kalian) di excel sebanyak 2x, enkripsi pertama menggunakan kunci pergeseran sesuai nomor absen. Selanjutnya hasil enkripsi pertama, kita enkripsi lagi menggunakan kunci dari kata "INFORMATIKA".

1. Kunci Geser : 19

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
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

Plaintext	Y	U	N	I	T	A	K	A	R	T	I	K	A	P	U	T	R	I
Ciphertext	R	N	G	B	M	T	D	T	K	M	B	D	T	I	N	M	K	B

2. Kunci : INFORMATIKA

Untuk mengatasi kelemahan chiper diatas maka perlu dilakukan modifikasi menggunakan kunci tanpa pengesaran.. yaitu menggunakan kata INFORMATIKA

INFORMATIKA→INFORMATK (Huruf yang sama harus dihilangkan)

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

Plaintext	R	N	G	B	M	T	D	T	K	M	B	D	T	I	N	M	K	B
Ciphertext	P	G	A	N	F	S	O	S	C	F	N	O	S	K	G	F	C	N

Penjelasan :

1. **Kunci Geser (Caesar Cipher) dengan Shift 19**

   Caesar Cipher adalah metode enkripsi klasik yang sangat sederhana, di mana setiap huruf pada **plaintext** digantikan dengan huruf lain berdasarkan pergeseran tetap dalam alfabet. Pada contoh ini, pergeseran yang digunakan adalah 19, yang berarti setiap huruf dalam **plaintext** digeser 19 posisi ke kiri dalam urutan alfabet

   Ketika kita melakukan pergeseran 19 posisi ke kiri, urutan alfabet menjadi:

1. Kunci Geser : 19

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

   Selanjutnya, setiap huruf pada **plaintext** akan diubah ke huruf yang sesuai pada urutan alfabet yang sudah digeser. Misalnya, jika **plaintext** yang akan dienkripsi adalah "Y U N I T A":

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

   Huruf **Y** akan diganti dengan **R**
   Huruf **U** akan diganti dengan **N**
   Huruf **N** akan diganti dengan **G**
   Huruf **I** akan diganti dengan **B**
   Huruf **T** akan diganti dengan **M**
   Huruf **A** akan diganti dengan **T**

   Hasil akhirnya, ciphertext yang dihasilkan dari teks **"Y U N I T A"** dengan kunci geser 19 adalah **"R N G B M T"**. **Kelemahan dari Caesar Cipher ini** adalah sifatnya yang sangat sederhana. Jika seseorang mengetahui bahwa metode Caesar Cipher digunakan dan mengetahui nilai pergeserannya (misalnya 19), mereka dapat dengan mudah mendekripsi pesan dengan menggeser huruf kembali ke posisi semula.

2. **Cipher dengan Kunci (Keyword Cipher) menggunakan Kata Kunci "INFORMATIKA"**

Untuk meningkatkan keamanan dari metode Caesar Cipher yang sederhana, kita dapat menggunakan pendekatan **Keyword Cipher**, yang memodifikasi alfabet berdasarkan kata kunci (keyword). Pada contoh ini, digunakan kata kunci **"INFORMATIKA"**.

Langkah pertama dalam metode ini adalah membentuk urutan alfabet baru berdasarkan kata kunci tersebut. Dimulai dengan menghilangkan huruf-huruf yang berulang dari kata kunci sehingga menjadi **"INFORMATK"**.
Setelah itu, sisa huruf dalam alfabet yang belum muncul di dalam kata kunci tersebut ditambahkan ke urutan, menghasilkan urutan alfabet baru:

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

Dalam metode ini, setiap huruf dari **plaintext** akan digantikan oleh huruf yang sesuai dalam urutan alfabet yang baru. Misalnya, jika kita ingin mengenkripsi teks **"R N G B M T D T K M B D T I N M K B"**, kita cukup mengganti setiap huruf dengan huruf yang sesuai dalam urutan alfabet baru:
Huruf **R** akan diganti dengan **P**
Huruf **N** akan diganti dengan **G**
Huruf **G** akan diganti dengan **A**
Huruf **B** akan diganti dengan **N**
Dan seterusnya.

Plaintext	R	N	G	B	M	T	D	T	K	M	B	D	T	I	N	M	K	B
Ciphertext	P	G	A	N	F	S	O	S	C	F	N	O	S	K	G	F	C	N

Hasil akhir dari enkripsi teks **"R N G B M T D T K M B D T I N M K B"** menggunakan kata kunci **"INFORMATIKA"** adalah **"P G A N F S O S C F N O S K G F C N"**.