

# Kamil Nowakowski

## 14150

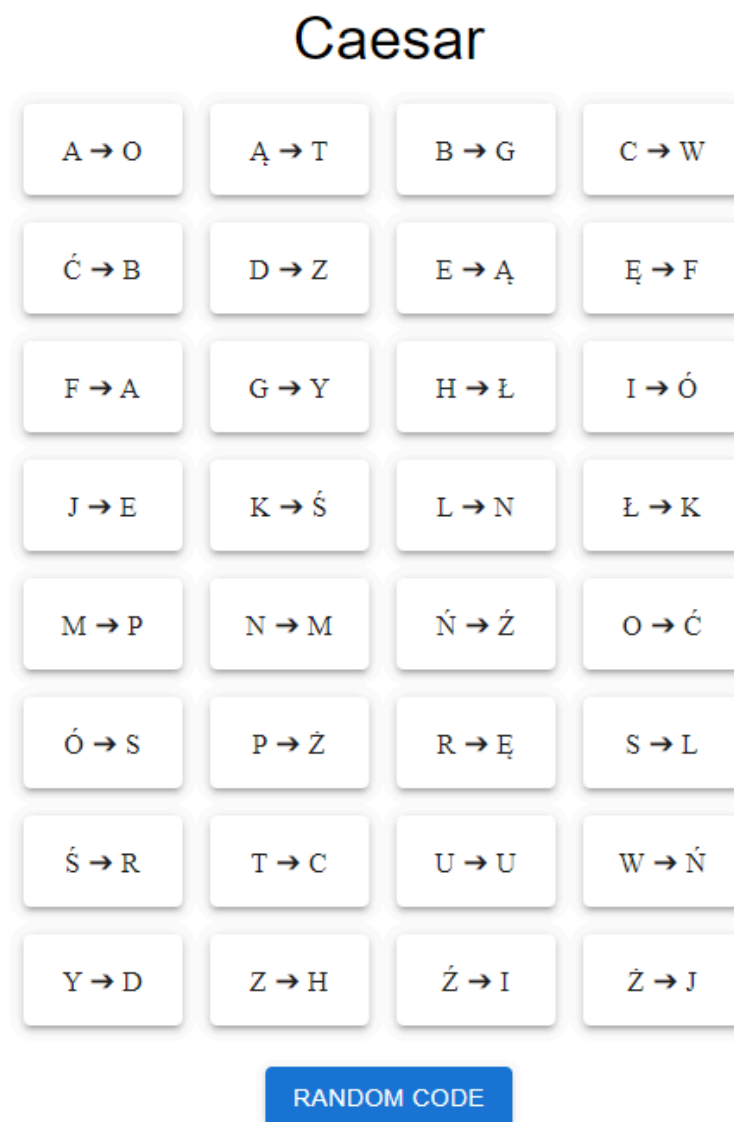
Projekt aplikacji internetowej zawierający proste funkcjonalne narzędzie do szyfrowania i deszyfrowania tekstu za pomocą klasycznej metody szyfrowania znanego jako **Szyfr Cezara**.

Link do repozytorium GitHub projektu:

[GitHub/KN/CipherCaesar](https://github.com/KN/CipherCaesar)

Aplikacja została zbudowana za pomocą języka programowania **TypeScript** używając technologii **React.js**, dodatkowo w zależnościach posiada zainstalowaną bibliotekę stylu **Material UI**.

**Widok panelu przedstawiającego aktualnie ustawiony szyfr.**



Poniżej znajduje się panel do zmieniania wartości zamienianych liter.

Y → D

Z → H

Ż → I

Ž → J

RANDOM CODE

Letter ▼

Value ▼

This letter will replaced

This value will replace the letter

SAVE EDIT

Wyświetlane są również alerty, gdy kod szyfru jest niekompletny.

Y → A

Z →

Ž →

Ž → F

! You must use all the letters!  
Missing elements: Ć, O, S

RANDOM CODE

Letter ▼

Value ▼

This letter will replaced

This value will replace the letter

SAVE EDIT

## Sekcja odpowiadająca za szyfrowanie i deszyfrację tekstu.

Y

This letter will be replaced

A

This value will replace the letter

SAVE EDIT

Provide the content to be encrypted

Przykładowa wiadomość do zaszyfrowania

ENCRIPTING

DECRYPTING

encrypted:

ŻOŚATJGFNBGBĄGFNRNDÓFNŚGIŚAŁONBGŃĄG

decrypted:

PRZYKŁADOWAWIADOMOŚĆDOZASZYFROWANIA

## Przykład deszyfracji.

Provide the content to be encrypted

ŻOŚATJGFNBGBĄGFNRNDÓFNŚGIŚAŁONBGŃĄG

ENCRIPTING

DECRYPTING

encrypted:

ŻOŚATJGFNBGBĄGFNRNDÓFNŚGIŚAŁONBGŃĄG

decrypted:

PRZYKŁADOWAWIADOMOŚĆDOZASZYFROWANIA

Kod zawiera klasę **"Encryption"** która ma klasę zagnieżdżoną **"Cipher\_Caesar"**, która jest klasą pochodną **"Cipher"**.

```
1  import Cipher from "./Cipher";
2  import CipherCaesar from "./Cipher_Caesar";
3
4  export default class Encryption {
5      private static instance: Encryption;
6      public text: string = "";
7      public encrypted: string = "";
8      public decrypted: string = "";
9
10     public crpher: Cipher;
11
12     constructor(crpher: Cipher) {
13         this.crpher = crpher
14     }
15
16     public static getInstance(): Encryption {
17         if (!Encryption.instance) {
18             Encryption.instance = new Encryption(new CipherCaesar());
19         }
20         return Encryption.instance;
21     }
22
23     public changeCipher(crpher: Cipher) {
24         this.crpher = crpher
25     }
26 }
```

```

1  import Alphabet from "../enum/enumAlphabet";
2  import shuffleArray from "../libs/shuffleArray";
3
4  export default abstract class Cipher {
5      public encryptAlphabet: string[] = []
6
7      public randomAlphabet(): string[] {
8          const alphabet = Object.values(Alphabet) as string[];
9          const randomAlphabet = shuffleArray(alphabet);
10         return randomAlphabet;
11     }
12
13     abstract decode(text: string): string;
14     abstract encrypt(text: string): string;
15 }
16

```

```

1  import Cipher from "../class/Cipher.ts";
2  import Alphabet from "../enum/enumAlphabet.ts";
3
4  export default class Cipher_Caesar extends Cipher {
5      private alphabet = Object.values(Alphabet) as string[];
6      public encryptAlphabet: string[] = [];
7
8  > public decode(text: string): string { ...
22 }
23
24 > public encrypt(text: string): string { ...
37 }
38 }
39

```

Widok całej aplikacji:

## Caesar

A → G	Ą → Y	B → K	C → W
Ć → Ó	D → F	E → S	Ę → U
F → L	G → E	H → P	I → Ą
J → H	K → T	L → Z	Ł → J
M → R	N → Ń	Ń → Ż	O → N
Ó → C	P → Ż	R → O	S → I
Ś → D	T → M	U → Ę	W → B
Y → A	Z → Ś	Ž → L	Ž → Ć

RANDOM CODE

Letter	Value
<div>Y</div>	<div>A</div>
This letter will be replaced	This value will replace the letter

SAVE EDIT

Provide the content to be encrypted

ŻOŚATJGFNBGBĄGFNRNDÓFNŚGIŚŁONBGŃĄG

ENCRYPTING

DECRYPTING

encrypted:

ŻOŚATJGFNBGBĄGFNRNDÓFNŚGIŚŁONBGŃĄG

decrypted:

PRZYKŁADOWAWIADOMOŚĆDOZASZYFROWANIA