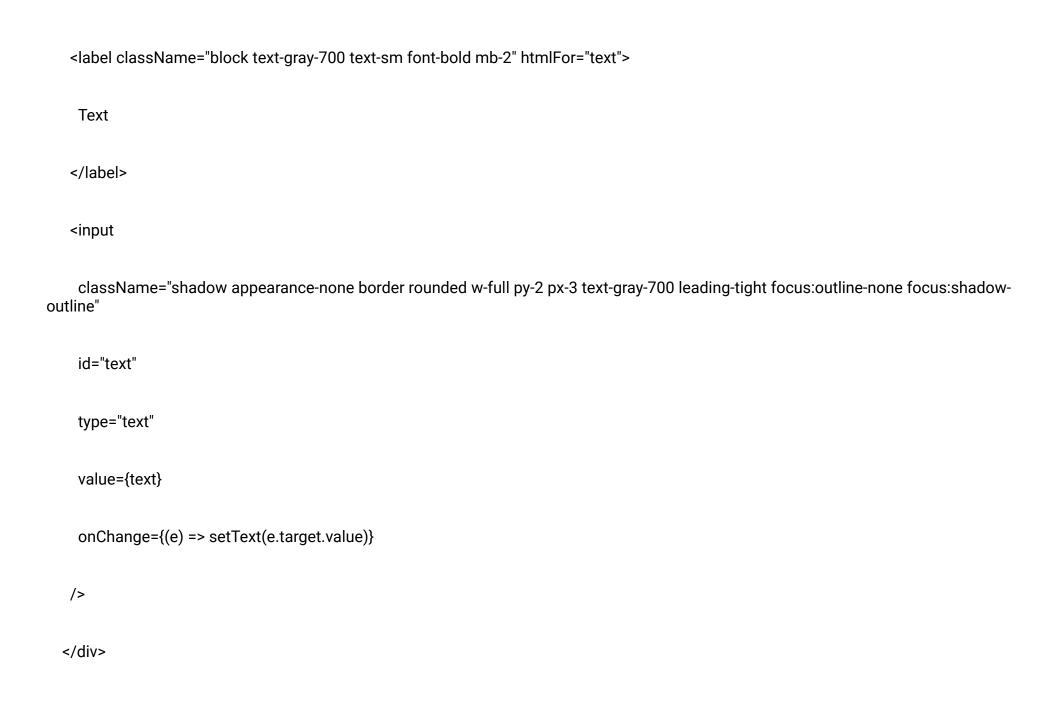
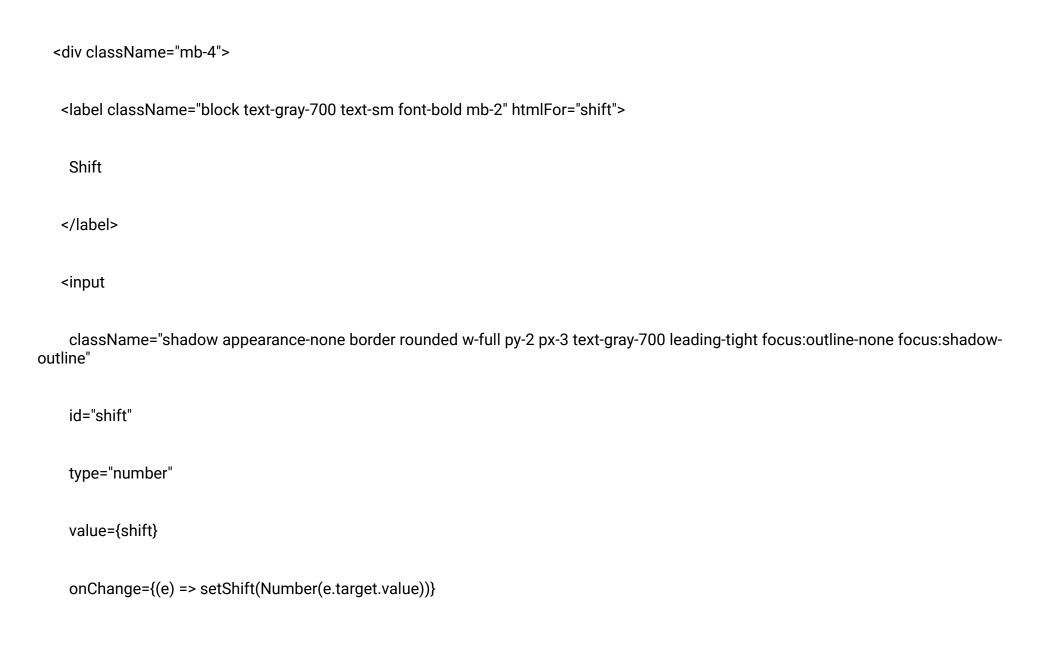
```
import { useState } from 'react';
interface CaeserCipherProps {}
const CaeserCipher: React.FC<CaeserCipherProps> = () => {
const [text, setText] = useState(");
const [shift, setShift] = useState(3);
const [encryptedText, setEncryptedText] = useState(");
const [decryptedText, setDecryptedText] = useState(");
const handleEncrypt = () => {
```

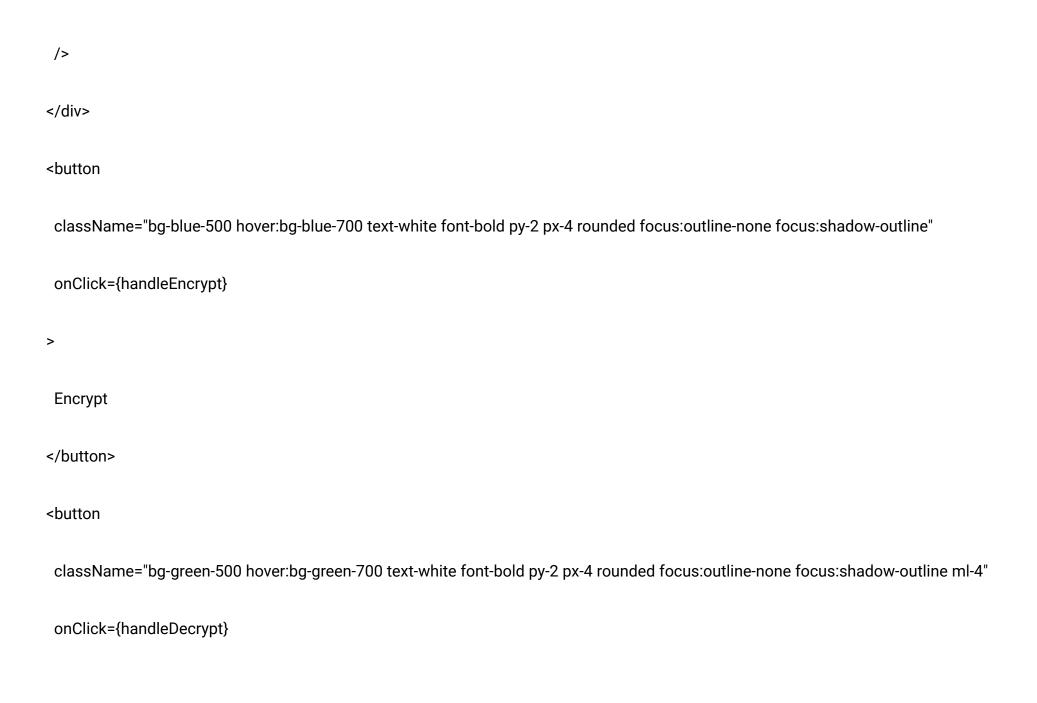
```
let encrypted = ";
for (let i = 0; i < text.length; i++) {
const charCode = text.charCodeAt(i);
if (charCode >= 65 && charCode <= 90) {
  encrypted += String.fromCharCode((charCode - 65 + shift) % 26 + 65);
} else if (charCode >= 97 && charCode <= 122) {
  encrypted += String.fromCharCode((charCode - 97 + shift) % 26 + 97);
} else {
  encrypted += text[i];
```

```
setEncryptedText(encrypted);
};
const handleDecrypt = () => {
 let decrypted = ";
 for (let i = 0; i < encryptedText.length; i++) {
  const charCode = encryptedText.charCodeAt(i);
  if (charCode >= 65 && charCode <= 90) {
   decrypted += String.fromCharCode((charCode - 65 - shift + 26) % 26 + 65);
  } else if (charCode >= 97 && charCode <= 122) {
   decrypted += String.fromCharCode((charCode - 97 - shift + 26) % 26 + 97);
```

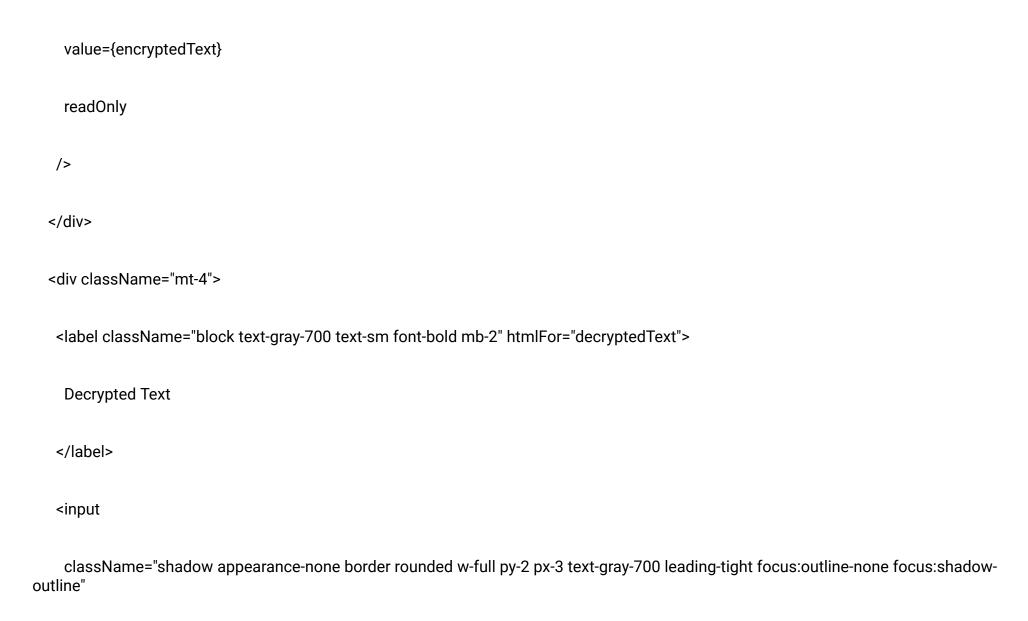
```
} else {
   decrypted += encryptedText[i];
 setDecryptedText(decrypted);
};
return (
 <div className="max-w-md mx-auto p-4 bg-white rounded-md shadow-md">
  <h1 className="text-2xl font-bold mb-4">Caeser Cipher</h1>
  <div className="mb-4">
```







```
>
   Decrypt
  </button>
  <div className="mt-4">
   <label className="block text-gray-700 text-sm font-bold mb-2" htmlFor="encryptedText">
    Encrypted Text
   </label>
   <input
    className="shadow appearance-none border rounded w-full py-2 px-3 text-gray-700 leading-tight focus:outline-none focus:shadow-
outline"
    id="encryptedText"
    type="text"
```



```
id="decryptedText"
  type="text"
  value={decryptedText}
  readOnly
 />
</div>
</div>
```

export default CaeserCipher;

};