



Coding School



```
1 <!DOCTYPE html>
2 <html lang="en">
3 <head>
4   <meta charset="UTF-8">
5   <meta http-equiv="X-UA-Compatible" content="IE=edge">
6   <meta name="viewport" content="width=device-width, initial-scale=1.0">
7   <link rel="stylesheet"
8     href="imageOpacity.css">
9   <title>Document</title>
10 </head>
11 <body>
12   <div class="demo-wrap">
13     
18     <div class="text">
19       <h1>Hello World!</h1>
20     </div>
```

html

Glitch



වෙබ් පිටුව



Glitch ආරම්භ කරන්න

- ❖ HTML CSS හා JS භාවිතයෙන් Captcha එකක් නිර්මාණය කරමු.



- ❖ ඉහත ආරම්භ කරන්න ලෙස ලබා දී ඇති link එක මගින් ව්‍යාපෘතිය ආරම්භ කරන්න.
- ❖ Body tag එක සඳහා පහත පරිදි CSS add කරගන්න.

```
body{  
    display: flex;  
    align-items: center;  
    justify-content: center;  
    min-height: 100vh;  
    background: #4db2ec;  
}
```

- ❖ Selection area එකෙහි CSS පහත පරිදි සකස් කරන්න.

```
::selection{  
    color: #fff;  
    background: #4db2ec;  
}
```

- ❖ wrapper ලෙස class name එක සහිත div tag එකක් body එක තුළ add කරගන්න. එය තුළ header tag එකක් තුළ පහත පරිදි heading එකක් නිර්මාණය කරන්න.

```
<div class="wrapper">
  <header>Captcha in JavaScript</header>
</div>
```

- ❖ එම elements සඳහා CSS පහත ආකාරයට වේ.

```
.wrapper{
  max-width: 485px;
  width: 100%;
  background: #fff;
  padding: 22px 30px 40px;
  border-radius: 10px;
  box-shadow: 8px 8px 8px rgba(0, 0, 0, 0.05);
}

.wrapper header{
  color: #4db2ec;
  font-size: 33px;
  font-weight: 500;
  text-align: center;
}
```

- ❖ දැන් captcha-img ලෙස නවත් div tag එකක් open කර එය තුළ පහත පරිදි img tag එකක් හා span tag එකක් නිර්මාණය කරගන්න.

```
<div class="captcha-area">
  <div class="captcha-img">
    </span>
  </div>
  <button class="reload-btn"><i class="fas fa-redo-alt"></i></button>
</div>
```

- ❖ Image එකට අදාළ file එක assets folder එක තුළ ඇත. එහි link එක copy කරගන src සඳහා ලබාදෙන්න.
- ❖ ඉහත elements සඳහා පහත ආකාරයට CSS add කරගන්න.

```

.wrapper .captcha-area{
  display: flex;
  height: 65px;
  margin: 30px 0 20px;
  align-items: center;
  justify-content: space-between;
}
.captcha-area .captcha-img{
  height: 100%;
  width: 345px;
  user-select: none;
  background: #000;
  border-radius: 5px;
  position: relative;
}

.captcha-img .captcha{
  position: absolute;
  left: 50%;
  top: 50%;
  width: 100%;
  color: #fff;
  font-size: 35px;
  text-align: center;
  letter-spacing: 10px;
  transform: translate(-50%, -50%);
  text-shadow: 0px 0px 2px #b1b1b1;
  font-family: 'Noto Serif', serif;
}

.captcha-area .captcha-img{
  height: 100%;
  width: 345px;
  user-select: none;
  background: #000;
  border-radius: 5px;
  position: relative;
}
.captcha-img img{
  width: 100%;
  height: 100%;
  object-fit: cover;
  border-radius: 5px;
  opacity: 0.95;
}

.wrapper button{
  outline: none;
  border: none;
  color: #fff;
  cursor: pointer;
  background: #4db2ec;
  border-radius: 5px;
  transition: all 0.3s ease;
}
.wrapper button:hover{
  background: #2fa5e9;
}
.captcha-area .reload-btn{
  width: 75px;
  height: 100%;
  font-size: 25px;
}

```

- ❖ දැන් input area එක නිර්මාණය සඳහා form එකක් සකස් කරමු.
පහත පරිදි form tag එකක් තුළ input tag එකක් හා button එකක්
නිර්මාණය කරගන්න.

```

<form action="#" class="input-area">
  <input type="text" placeholder="Enter captcha" maxlength="6" spellcheck="false" required>
  <button class="check-btn">Check</button>
</form>

```

මෙහි input area එකෙහි ඇතුළත් කළ හැකි උපරිම අගය 6 වන
ලෙස සකස් කිරීමට maxlength attribute එක භාවිතා කර ඇත.

- ❖ ඉහත elements සඳහා CSS මේ ආකාරයට සකස් කරගන්න.


```

.input-area input{
  width: 100%;
  height: 100%;
  outline: none;
  padding-left: 20px;
  font-size: 20px;
  border-radius: 5px;
  border: 1px solid #bfbfbf;
}
.input-area input:is(:focus, :valid){
  padding-left: 19px;
  border: 2px solid #4db2ec;
}

.input-area input:valid + .check-btn{
  opacity: 1;
  pointer-events: auto;
}
.wrapper .status-text{
  display: none;
  font-size: 18px;
  text-align: center;
  margin: 20px 0 -5px;
}

.input-area input::placeholder{
  color: #bfbfbf;
}
.input-area .check-btn{
  position: absolute;
  right: 7px;
  top: 50%;
  font-size: 17px;
  height: 45px;
  padding: 0 20px;
  opacity: 0;
  pointer-events: none;
  transform: translateY(-50%);
}

```

- ❖ ඒ ආකාරයට design එක අවසන් කරගත් පසු JavaScript මගින් captcha එක නිවැරදිව වැඩ කරන ආකාරයට සකස් කරමු.
- ❖ ඒ සඳහා script.js file එකට ගොස් පහත ආකාරයට variables 5ක් නිර්මාණය කර ඒවාට querySelector මගින් class name හරහා html elements add කරගන්න.

```

const captcha = document.querySelector(".captcha"),
reloadBtn = document.querySelector(".reload-btn"),
inputField = document.querySelector(".input-area input"),
checkBtn = document.querySelector(".check-btn"),
statusTxt = document.querySelector(".status-text");

```

- ❖ තවත් variable එකක් නිර්මාණය කර එයට ඉංග්‍රීසි හෝඩියේ සියලුම capital හා simple අකුරු 0 සිට 9 තෙක් ඉලක්කම් array එකක් ලෙස ලබා දෙන්න.

```
let allCharacters = ['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', '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', 0, 1, 2, 3, 4, 5, 6, 7, 8, 9];
```

- ❖ දැන් random letters හා numbers combination එකක් ලෙස captcha එකක් නිර්මාණය කර එය “captcha” නැමැති class name එක ඇති span tag එකෙහි text එක ලෙස පෙන්වීමට පහත ආකාරයට function එකක් නිර්මාණය කරන්න.

```
function getCaptcha(){
  for (let i = 0; i < 6; i++) {
    let randomCharacter = allCharacters[Math.floor(Math.random() * allCharacters.length)];
    captcha.innerHTML += ` ${randomCharacter}`;
  }
}
```

- ❖ එයට පහළින් එම function එක call කරන්න. එවිට web site එක load වන විටම captcha එකක් දර්ශනය වේ.

```
getCaptcha();
```

- ❖ සියල්ල clear කර දැමීමට අදාළව removeContent ලෙස function එකක් නිර්මාණය කරගත යුතුය. එහි inputField variable එකෙහි, captcha span tag එකෙහි values empty කර statusTxt element එකෙහි display style එක none කර hide කිරීම සිදු කරයි.

```
function removeContent(){
  inputField.value = "";
  captcha.innerHTML = "";
  statusTxt.style.display = "none";
}
```

- ❖ ඉන්පසු reload button එක මත click කළ විට සිදු විය යුතු දේ නිර්මාණයට eventListener එකක් පහත පරිදි සකස් කරන්න.

```
reloadBtn.addEventListener("click", ()=>{
    removeContent();
    getCaptcha();
});
```

- ❖ එහිදී ඉහත නිර්මාණය කරගත් removeContent හා getCaptcha functions දෙක call කිරීම සිදු කරයි.
- ❖ අවසාන වශයෙන් user enter කරන ලද input එක හා captcha එක නිවැරදි දැයි check කර status එක දර්ශනය වීම නිර්මාණය කරගැනීමට checkBtn එකට පහත පරිදි eventListener එකක් code කරමු.

```
checkBtn.addEventListener("click", e =>{
    e.preventDefault();
    statusTxt.style.display = "block";
    let inputVal = inputField.value.split('').join(' ');
    if(inputVal == captcha.innerText){
        statusTxt.style.color = "#4db2ec";
        statusTxt.innerText = "Nice! You don't appear to be a robot.";
        setTimeout(()=>{
            removeContent();
            getCaptcha();
        }, 2000);
    }else{
        statusTxt.style.color = "#ff0000";
        statusTxt.innerText = "Captcha not matched. Please try again!";
    }
});
```

මෙහිදී inputVal variable එකට දමාගන්නා user input එක captcha variable එකෙහි value එකට සමාන දැයි check කර සමාන නම් statusTxt එකෙහි value එක ලෙස “Nice! You don’t appear to be a ronot.”

ලෙසින් එම අගයන් දෙක match නොවේ නම් “Captcha not matched. Please try again” ලෙසින් දර්ශනය වේ. ඊට අමතරව removeContent function එක call වීම සිදු වී සියල්ල clear වී getCaptcha function එක මගින් අලුත් pattern එක්ක load වීම සිදු වේ.

Captcha in JavaScript





Nice! You don't appear to be a robot.

Captcha in JavaScript





Captcha not matched. Please try again!