

SOURCE CODE: index.html

```
<!DOCTYPE html>
<html lang="en">
  <head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Password Generator</title>

    <!--Stylesheets-->
    <link rel="stylesheet" type="text/css" href="style.css" />
    <link rel="stylesheet"
href="https://fonts.googleapis.com/css2?family=Material+Symbols+Outlined:opsz,wght,FILL,GRAD@20..48,100..700,0..1,-50..200" />
    <link rel="stylesheet"
href="https://fonts.googleapis.com/css2?family=Material+Symbols+Outlined:opsz,wght,FILL,GRAD@20..48,100..700,0..1,-50..200" />
  </head>

  <body>
    <div id="content_1">
      <h2>Password Generator</h2>

      <div id="password_container">
        <div id="password_div"><span id="password"></span></div>
        <button id="clipboard">
          <span style="font-size: 27px" class="material-symbols-outlined">file_copy</span>
        </button>
      </div>
    </div>
  </body>
</html>
```

```
<div id="custom_container">
  <div id="length">
    <label for="range_length">
      Password Length:
      <input type="range" id="range_value" min="8" max="20" value="8">
    </label>

    <span id="display_range">08</span>
  </div>

  <div class="checkbox_container">
    <label class="checkbox">
      <input type="checkbox" id="uppercase" checked>
      ABC
    </label>

    <label class="checkbox">
      <input type="checkbox" id="lowercase" checked>
      abc
    </label>

    <label class="checkbox">
      <input type="checkbox" id="numbers" checked>
      123
    </label>

    <label class="checkbox">
```

```
        <input type="checkbox" id="symbols" checked>
        !@%
    </label>
</div>

<div id="button_container_1">
    <button id="generate_password">Generate Password</button>
    <button id="view_history">View History</button>
</div>

    <p id="label">Already have a password? Then try the <span
id="view_strength_meter">Strength Meter</span></p>
</div>
</div>

<div id="content_2">
    <div id="history">
        <div class="content_title">
            <h2>View History</h2>
            <button id="hide_history">X</button>
        </div>

        <div class="history_table">
            <table id="table">
                <colgroup>
                    <col>
                    <col>
                    <col>
```

```

        </colgroup>

        <tr>
            <th>Date</th>
            <th>Time</th>
            <th>Password</th>
        </tr>
    </table>
</div>

    <button id="clear_history">Clear History</button>
</div>
</div>

<div id="content_3">
    <div id="strength_meter_container">
        <div class="content_title">
            <h2>Strength Meter</h2>
            <button id="hide_strength_meter">X</button>
        </div>

        <div id="form_control">
            <div id="strength_meter">
                <input type="password" id="password_sample" placeholder="Type your password here">
                <button id="check_password">
                    <span style="font-size: 30px;" class="material-symbols-outlined">check</span>
                </button>
            </div>
        </div>
    </div>

```

```
        <span id="strength_status"></span>
        <label for="show_password">
            <input type="checkbox" id="show_password">
            Show Password
        </label>
    </div>

    <p id="result">Your password must have:</p>
    <ul>
        <li id="password_length">Length of 8-20 characters</li>
        <li id="password_letters">Uppercase and/or Lowercase Letters</li>
        <li id="password_numbers">Numbers</li>
        <li id="password_symbols">Special Characters</li>
    </ul>
</div>
</div>

<!--JavaScript FIle-->
    <script src="script.js"></script>
</body>
</html>
```

SOURCE CODE: style.css

```
@import url('https://fonts.googleapis.com/css2?family=Source+Code+Pro:wght@300;500&display=swap');

* {
  margin: 0px;
  padding: 0px;
  font-family: 'Source Code Pro', monospace;
}

body {
  background-color: #FFF8D6;
}

#content_1 {
  position: absolute;
  top: 50%;
  left: 50%;
  transform: translate(-50%, -50%);
  border-radius: 10px;
  box-shadow: 5px 5px #dbdbdb;
  height: auto;
  width: 430px;
  padding: 20px;
  background-color: #ffffff;
}

#password_container, #strength_meter {
  margin-top: 5px;
}
```

```
width: auto;
display: flex;
flex-direction: row;
align-items: flex-start;
justify-content: flex-start;
}

#length {
display: flex;
flex-direction: row;
}

#display_range {
margin-left: 15px;
}

#password_div, #password_sample {
width: 360px;
height: 18px;
border: 2px solid #5a5a5a;
padding: 10px;
font-size: 16px;
}

#clipboard, #check_password {
height: 42px;
width: 45px;
border: 2px solid #5a5a5a;
```

```
    background-color: #A4D0A4;
}

#custom_container {
    margin-top: 25px;
    display: flex;
    flex-direction: column;
}

.checkbox_container {
    display: flex;
    flex-direction: row;
    justify-content: space-between;
    width: 325px;
}

#button_container_1 {
    margin-top: 10px;
    display: flex;
    flex-direction: row;
}

#generate_password {
    border: 0px;
    border-radius: 5px;
    height: 35px;
    width: 190px;
    background-color: #A4D0A4;
```



```
}

#view_history {
    margin-left: 15px;
    border: 0px;
    border-radius: 5px;
    height: 35px;
    width: 145px;
    background-color: #F7E1AE;
}

#clipboard:hover, #generate_password:hover, #check_password:hover {
    background-color: #81e281;
    transition: 0.5s;
}

#view_history:hover {
    background-color: #ffda82;
    transition: 0.5s;
}

label, div, button {
    font-size: 16px;
}

#label {
    margin-top: 40px;
    text-align: center;
}
```

```
    font-size: 16px;
}

#view_strength_meter {
    color: #193bac;
}

#view_strength_meter:hover {
    cursor: pointer;
    text-decoration: underline;
}

#content_2, #content_3 {
    display: none;
    position: fixed;
    top: 0;
    left: 0;
    right: 0;
    bottom: 0;
    background-color: rgba(0, 0, 0, 0.6);
}

#content_2.show_content, #content_3.show_content {
    display: block;
}

#history {
    display: flex;
```

```
flex-direction: column;
position: absolute;
top: 50%;
left: 50%;
transform: translate(-50%, -50%);
border-radius: 10px;
height: 350px;
width: 635px;
background-color: #ffffff;
padding: 20px;
animation-name: open_container;
animation-duration: 0.3s;
}

.content_title {
margin-bottom: 10px;
display: flex;
flex-direction: row;
justify-content: space-between;
}

#hide_history, #hide_strength_meter {
margin-bottom: 10px;
height: 32px;
width: 32px;
border: none;
border-radius: 30px;
background-color: #d8d8d8;
```

```
    text-align: center;
    font-size: 20px;
}

#hide_history:hover {
    background-color: #bebebe;
    transition: 0.5s;
    cursor: pointer;
}

.history_table {
    overflow: auto;
    margin-bottom: 20px;
    align-content: center;
    height: 300px;
}

table {
    border-spacing: 0;
}

col:first-child {
    width: 170px;
}

col:nth-child(2) {
    width: 120px;
}
```

```
col:nth-child(3) {  
    width: 340px;  
}  
  
th, td {  
    text-align: center;  
    border: 1px solid #5a5a5a;  
}  
  
th {  
    background-color: #8cdf8c;  
}  
  
td {  
    background-color: #f0f0f0;  
}  
  
td:hover {  
    background-color: #dddddd;  
    cursor: context-menu;  
}  
  
#clear_history {  
    border: 0px;  
    border-radius: 5px;  
    height: 35px;  
    width: 180px;
```

```
    background-color: #ff8080;
}

#clear_history:hover {
    background-color: #ff6666;
    transition: 0.5s;
}

#strength_meter_container {
    display: flex;
    flex-direction: column;
    position: absolute;
    top: 50%;
    left: 50%;
    transform: translate(-50%, -50%);
    border-radius: 10px;
    height: auto;
    width: auto;
    padding: 20px;
    background-color: #ffffff;
    animation-name: open_container;
    animation-duration: 0.3s;
}

#form_control {
    margin-bottom: 15px;
    display: flex;
    flex-direction: column;
```

```
}

#password_sample:focus {
    outline: none;
}

#strength_status {
    font-size: 14px;
}

#result {
    margin-top: 15px;
    font-size: 16px;
}

li {
    margin-left: 30px;
    font-size: 14px;
}

.valid {
    visibility: visible;
    color: #00d123;
}

.invalid {
```

```
visibility: visible;
color: #ff2a2a;
}

@keyframes open_container {
  from {
    opacity: 0;
  }

  to {
    opacity: 1;
  }
}
```


SOURCE CODE: script.js

```
//Displaying the value of the input range
var range_length = document.getElementById("range_value");
var range_value = document.getElementById("display_range");

range_length.addEventListener('input', function() {
    range_value.textContent = range_length.value < 10 ? "0" + range_length.value : range_length.value;
});

//Show History
var view_history = document.getElementById("view_history");
var content_2 = document.getElementById("content_2");
view_history.addEventListener('click', () => content_2.classList.add('show_content'));

//Hide History
var hide_history = document.getElementById("hide_history");
hide_history.addEventListener('click', () => content_2.classList.remove('show_content'));
window.addEventListener('click', e => e.target == content_2 ?
content_2.classList.remove('show_content') : false);

//Show Strength Meter
var view_strength_meter = document.getElementById("view_strength_meter");
var content_3 = document.getElementById("content_3");
view_strength_meter.addEventListener('click', () => content_3.classList.add('show_content'));
```

//Empty Strength Meter

```
function empty_strength_meter() {  
    content_3.classList.remove('show_content');  
    password_sample.value = "";  
    show_password.checked = false;  
    password_sample.type = show_password.checked ? "text" : "password";  
    display_status.className = "";  
    display_status.innerHTML = "";  
    password_length.className = "";  
    password_letters.className = "";  
    password_numbers.className = "";  
    password_symbols.className = "";  
}
```

//Hide Strength Meter

```
var hide_strength_meter = document.getElementById("hide_strength_meter");  
hide_strength_meter.addEventListener('click', () => {  
    empty_strength_meter();  
});  
window.addEventListener('click', e => e.target == content_3 ? empty_strength_meter() : false);
```

//Generate Password

```
function generated_password() {  
    var uppercase_checkbox = document.getElementById("uppercase");  
    var lowercase_checkbox = document.getElementById("lowercase");  
    var number_checkbox = document.getElementById("numbers");  
    var symbol_checkbox = document.getElementById("symbols");
```

```
let charset = "";
if (uppercase_checkbox.checked) {
    charset += "ABCDEFGHIJKLMNOPQRSTUVWXYZ";
}
if (number_checkbox.checked) {
    charset += "1234567890";
}
if (lowercase_checkbox.checked) {
    charset += "abcdefghijklmnopqrstuvwxyz";
}
if (symbol_checkbox.checked) {
    charset += "!@#$%^&*()-_`~|'/?/";
}

let password = "";
for (let i = 1; i <= range_length.value; i++) {
    var randomIndex = Math.floor(Math.random() * charset.length);
    password += charset.charAt(randomIndex);
}

return password;
}
```

//Insert Data

```
function insert_data(generated_password) {
    //Get the current date
```

```

        var month = (new Date().getMonth() + 1) < 10 ? "0" + (new Date().getMonth() + 1) : new
Date().getMonth() + 1;
        var day = new Date().getDate() < 10 ? "0" + new Date().getDate() : new Date().getDate();
        var year = new Date().getFullYear();
        var date = month + "/" + day + "/" + year;

        //Get current time
        var hours = new Date().getHours() < 10 ? "0" + new Date().getHours() : new Date().getHours();
        var minutes = new Date().getMinutes() < 10 ? "0" + new Date().getMinutes() : new
Date().getMinutes();
        var seconds = new Date().getSeconds() < 10 ? "0" + new Date().getSeconds() : new
Date().getSeconds();
        var time = hours + ":" + minutes + ":" + seconds;

        //Creating a data array and return that data to use for history_arr
        var data = [date, time, generated_password];
        return data;
    }

    //Create History
    const table_body = document.getElementById("table");
    function create_history(history_arr) {
        history_arr.forEach(data => {
            const row = table_body.insertRow();

            data.forEach(value => {
                const cell = row.insertCell();
                cell.textContent = value;
            });
        });
    }

```

```
    });  
  });  
}
```

```
//Display password and history
```

```
var display_password = document.getElementById("password");  
var generate_password = document.getElementById("generate_password");
```

```
generate_password.addEventListener('click', () => {  
  var password = generated_password();  
  display_password.innerHTML = password;
```

```
  //Inserting the data array to the history_arr
```

```
    var data = insert_data(password);  
    var history_arr = [];  
    history_arr.push(data);  
    create_history(history_arr);
```

```
});
```

```
//Copy the password to clipboard
```

```
var clipboard = document.getElementById("clipboard");  
clipboard.addEventListener('click', () => {  
  var passwordText = document.getElementById("password");  
  var textArea = document.createElement("textarea");  
  textArea.value = passwordText.textContent;  
  document.body.appendChild(textArea);  
  textArea.select();
```

```
document.execCommand("Copy");
document.body.removeChild(textArea);
alert("The generated password has been copied on the clipboard");
});
```

//Clear History

```
var clear_history = document.getElementById('clear_history');
clear_history.addEventListener('click', () => {
    while (table_body.rows.length > 0) {
        table_body.deleteRow(1);
    }
});
```

//Show Password

```
var show_password = document.getElementById("show_password");
show_password.addEventListener('change', () => {
    password_sample.type = show_password.checked ? "text" : "password";
});
```

//Check for Special Characters

```
function check_symbols(password_sample) {
    var symbols = /[!@#$%^&*(),.?"':{}|<>]/;
    return symbols.test(password_sample);
}
```

```
//Display Strength Meter Status
var check_password = document.getElementById("check_password");
var display_status = document.getElementById("strength_status");
var password_length = document.getElementById("password_length");
var password_letters = document.getElementById("password_letters");
var password_numbers = document.getElementById("password_numbers");
var password_symbols = document.getElementById("password_symbols");

//Main counter that will increase if the user has a specific character on their password
let main_counter = 0;

check_password.addEventListener('click', () => {
    //Main counter will reset for another password checking
    main_counter = 0;
    var password_sample = document.getElementById("password_sample");

    //Check if the password has symbols
    if (check_symbols(password_sample.value)) {
        password_symbols.classList.remove("invalid");
        password_symbols.classList.add("valid");
        main_counter++;
    } else {
        password_symbols.classList.remove("valid");
        password_symbols.classList.add("invalid");
    }

    //Check if the password has upper/lower case letters
    var lowerCaseLetters = /[a-z]/g;
```

```
        var upperCaseLetters = /[A-Z]/g;
        if(password_sample.value.match(lowerCaseLetters) ||
password_sample.value.match(upperCaseLetters)) {
            password_letters.classList.remove("invalid");
            password_letters.classList.add("valid");
            main_counter++;
        } else {
            password_letters.classList.remove("valid");
            password_letters.classList.add("invalid");
        }

//Check if the password has numbers
        var numbers = /[0-9]/g;
        if(password_sample.value.match(numbers)) {
            password_numbers.classList.remove("invalid");
            password_numbers.classList.add("valid");
            main_counter++;
        } else {
            password_numbers.classList.remove("valid");
            password_numbers.classList.add("invalid");
        }

//Check if the password has more than 8 characters
        if(password_sample.value.length >= 8) {
            password_length.classList.remove("invalid");
            password_length.classList.add("valid");
        } else {
            password_length.classList.remove("valid");
```



```
        password_length.classList.add("invalid");
    }

    //Display the password strength status
    if(main_counter <= 2 || password_sample.value.length <= 7) {
        display_status.classList.remove("valid");
        display_status.classList.add("invalid");
        display_status.innerHTML = "Password is not strong";
    } else {
        display_status.classList.remove("invalid");
        display_status.classList.add("valid");
        display_status.innerHTML = "Password is strong";
    }

});
```