

Friday, Jan 19<sup>th</sup> 2024. [Building a registration form].

- ① Adding `<!DOCTYPE html>`
- ② Adding `<html lang="en"></html>`
- ③ Adding opening & closing head & body elements within the html element.
- ④ Adding `<meta charset="utf-8">` and `<title> Registration Form </title>`.
- ⑤ Adding `<link rel="stylesheet" href="styles.css">` within the head element.
- ⑥ Adding h1 element within the body element. `<h1> Registration Form </h1>`.
- ⑦ Adding paragraph `<p></p>`. → "Please fill..."
- ⑧ (styles.css) giving the body width: 100%, and height: 100vh.

```
body {  
    width: 100%;  
    height: 100vh;  
}
```

# vh unit stands for viewport height, and is relative  
to 1% of the height of the viewport. #

- ⑨ (styles.css) setting the body's margin into 0 (zero).
- ⑩ (styles.css) changing the body background-color to `#1B1B32` and the  
text color `#F5F6F7`.  
`background-color: #1B1B32;  
color: #F5F6F7;`
- ⑪ (index.html) inserting a form with an action attribute targeting a link. (It is  
done after inserting a paragraph (see step 7.).  
`<form action="(link)"></form>`
- ⑫ (index.html) adding method attribute. It specifies how to send form-data  
to the URL specified in the action attribute. The form-data can be sent via  
a GET request as URL parameters (with `method="get"`) or via a POST  
request as data in the request body (with `method="post"`).  
`<form action="(link)" method="POST"></form>`
- ⑬ (index.html) adding three `<fieldset></fieldset>` within the form element.
- ⑭ (index.html) adding four label elements within the 1<sup>st</sup> fieldset.
- ⑮ (index.html) adding the texts to the label elements.
- ⑯ (styles.css) adding:

```
label {  
    display: block;  
    margin: 0.5rem 0;  
}
```

# "rem" stands for root em and is relative  
to the font size of the html element #

3

17 (index.html) nesting an `input` element within each `label`.

`<label>XXX: <input></label>`

Note: `input` element is self-closing element.

18 (index.html) [this seems hard, so remember the format pls!!]

Linking the `input` elements & the `label` elements together using `for` attribute.

`<label for="first-name"> Enter Your First Name: <input id="first-name"/></label>`  
etc.

# The value of `for` and `id` should be same #.

19 (index.html) specifying the `type` attribute.

`<label for="first-name"> Enter Your First name: <input id="first-name" type="text"></label>`

ini buat nama (teks)

kalau buat email ya "email",  
password ya "password".

20 (index.html)

To handle the form submission, after the last `fieldset` element, add an `input` element with the `type` attribute set to `submit` and the `value` attribute set to `Submit`.

`<input type="submit" value="Submit">`

21 (index.html) Making the form more interactive by adding `required` attribute.

for example:

`<label for="abc"> Xyz: <input id="abc" type="text" required/></label>`

22 (index.html) adding custom validation to the `password` `input` element, by adding a `minlength` attribute with a value of 8.

Note: certain `type` attribute values come with built-in form validation. For example → `type = "email"` requires that the value be a valid email address.

`<label for="new-password"> Create a New Password: <input id="new-password" type="password" required minlength="8"/></label>`

23 (index.html) # with `type = "password"`, you can use the `pattern` attribute to define a regular expression that the password must match to be considered valid.

# add a `pattern` attribute to the `password` `input` element to require the input match: `[a-zA-Z]{8,}` → regular expression which matches eight or more lowercase letters `a` to `z` or the digits `0` to `9`. Then, remove the `minlength` attribute & try it out.

(see step 22) `minlength = "8"` diganti jadi `pattern = "[a-zA-Z]{8,}"`

24 (index.html) adding 2 `label` elements inside the second `fieldset`.

25 (index.html) adding `input` element with `type = "radio"` within these 2 `label` elements [see step 24].

(26) (index.html) `<label><input type="radio"> Personal </label>`  
`<label><input type="radio"> Business </label>`

add space here!

(27) (index.html) # to relate the radio inputs, give them the same name attribute with a value of `account-type`. Now, it is impossible to select both radio inputs at the same time.

`<label><input type="radio" name="account-type"/> Personal </label>`  
`<label><input type="radio" name="account-type"/> Business </label>`

(28) (index.html) adding a legend element to substitute required attribute.  
[ soalnya kalau pakai `required` malah gabisa dipake untuk tipe radio button ].  
[ nambahinnya di dalam fieldset ya. Trus jangan lupa kasih attribute `checked` next to personal input.]

`<fieldset>`  
`<legend>Account type (required)</legend>`  
`<label><input type="radio" name="account-type" checked/> Personal </label>`  
`<label><input type="radio" name="account-type"/> Business </label>`  
`</fieldset>`

(29) (index.html) Following the accessibility by linking the input elements and the label.

→ changed into like this:

`<label for="personal-account">`  
trus di dalam input tambahin `id="personal-account"`  
`<label for="business-account">`  
trus di dalam input tambahin `id="bu"`  
business-account



STEP 28 & 29  
ARE FIXED FORMS

(30) (index.html) adding this below

step 29 :

`<label for="terms-and-conditions"><input type="checkbox" id="terms-and-conditions" required></label>`

(31) add here : I accept the `<a href="..."> terms and conditions`

(32) (index.html) Allowing a user to upload a profile picture.

`<fieldset>`  
`<label>Upload a profile picture: <input type="file"></label>`  
`</fieldset>`

(33) (index.html)

`<label>Input your age (years): <input type="number" min="13" max="120"></label>`

③4 (index.html) Adding a dropdown to the form.  
# start by adding a select element then nest 5 option elements within the select element.

```
<select>
  <option></option> x5
</select>
```

③5 (index.html) adding this:

```
<label> How do you hear about us? </label>
```

③6 (index.html) adding text to each element.

③7 (index.html) adding value to each option.

```
<select>
  <option value="">(select one)</option>
  <option value="1">abc </option>
  <option value="2">def </option>
  <option value="3">ghi </option>
</select>
```

③8 (index.html) The **textarea** element acts like an **input** element of type text, but comes with the added benefit of being able to receive multi-line text, and an initial number of text rows & columns.

# users will be able to register with a bio. Add a label with the text **Provide a bio:** at the end of the **fieldset**. Add a **textarea** element inside the label.

```
<label> Provide a bio: <textarea></textarea></label>.
```

③9 (index.html) linking the applicable form elements and their **label** elements. (see screenshot image on tablet's gallery !!.) [or see step 29 for reference].

⑩ (index.html) adding **rows** (rows) and **cols** (columns) attributes within **textarea** element.

⑪ (index.html) giving the **textarea** a **placeholder** (biar orang ada ide, kayak ditulis prompt gitu).

↓  
attribute

```
<textarea id="bio" rows="3" cols="30" placeholder="I love you..."></textarea>
```

⑫ (index.html) adding unique name attribute (up to you)  
ex :

```
<label for="first-name">Enter Name: <input id="first-name" type="text" required  
name="abc"/></label>
```

Value untuk name attribute terserah kita.

④3 (styles.css) changing the pretelan like fonts, size.

```
body {  
    width: 100%;  
    height: 100vh;  
    margin: 0;  
    margin: 0;  
    background-color: # FFFFFF;  
    color: # ABCDEF;  
    font-family: Tahoma;  
    font-size: 16px;  
}
```

④4 (styles.css) # center the h1 & p elements by giving them a margin of 1em auto. Then, align their text in the center.

```
h1, p {  
    margin: 1em auto;  
    text-align: center;  
}
```

④5 (styles.css)

```
fieldset {  
    padding-top: 2rem;  
    padding-bottom: 2rem;  
    padding-left: 0px;  
    padding-right: 0px;  
    border: 0px;  
}
```

④6 (styles.css) using the last-of-type CSS pseudo-class like this:

```
p:last-of-type {} → fieldset:last-of-type {  
    border-bottom: none;  
}
```

④7 (styles.css)

```
input, textarea, select {  
    margin-top: 10px;  
    margin-bottom: 0px;  
    margin-left: 0px;  
    margin-right: 0px;  
    width: 100%;  
}
```

}

④8 (styles.css)

```
• inline {  
    width: unset;
```

}

④9 modifying the form element, by adding margin & width.

```
form {  
    margin: 0 auto;  
    max-width: 500px;  
    min-width: 300px;  
    width: 60vw;
```

(styles.css)

⑤0 changing step 46 into :

```
fieldset {  
    border: none;  
    padding: 2rem 0;  
    border-bottom: 3px solid #3B3B4F;  
}
```

(styles.css)

⑤1 Di fieldset yang ada account type, dalam label input id ditambahin class="inline" di label personal & business account.

[index.html]

52) Adding some space between .inline elements and the label text.  
(styles.css)

.inline {

```
width: unset;  
margin-top: 0;  
margin-bottom: 0;  
margin-right: 0.5em;  
margin-left: 0;
```

}

ATAU → margin: 0 0 0.5 0;

53) (styles.css) set vertical-align: middle; for .inline.

54) (styles.css) blending the color for input & textarea.

input, textarea {

```
background-color: #0a0a23;  
border: 1px solid #0a0a23;
```

55) color: #FFFFFF;

min-height: 2em;

}

56) (styles.css) mindahin min-height: 2em; ke bagian .select di step 51.

57) (styles.css) adding:

input [type = "submit"] {

```
display: block;  
width: 60%;
```

}

58) nambahin margin: 0 auto; di step 57.

59) nambahin

```
height: 2em;  
font-size: 1.1rem;
```

60) (styles.css) nambahin:

```
background-color: #3B3B4F;  
border-color: white;
```

61) (styles.css) adding min-width: 300px;

62) input [type = "file"] {

padding: 1px 2px;

}

63) form {

```
width: 60vw;  
max-width: 500px;  
min-width: 300px;  
margin: 0 auto;  
padding-bottom: 2em;
```

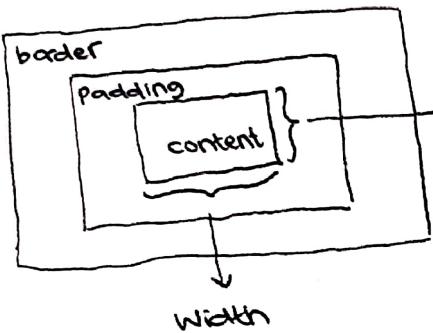
}

- (GA) (index.html) adding `class = "inline"` in `<input id = "terms-and-conditions" type = "checkbox" required name = "terms-and-conditions" class = "inline">`  
(GS) (styles.css) changing the color.

Tuesday, 23<sup>rd</sup> Feb 2024

Learn the CSS Box Model by Building a Rothko Painting. :)

- ① adding DOCTYPE, html element, head, & body
- ② the order: inside the `<head>`, add `<meta charset = ...>` and `<title>`.  
inside the `<body>`, add `<img src = "...>`.
- ③ changing the src attribute.



- ④ # bagian terluar setelah border = MARGIN !
- ⑤ Removing the `<img>` element.
- ⑥ `<div class = "canvas"></div>`

- ⑦ Linking the index.html & styles.css.
- ⑧ (styles.css) setting the `.canvas` width → (500px)
- ⑨ (styles.css) setting the `.canvas` height → (600px)
- ⑩ (styles.css) changing the background color of the canvas → #4d0f00
- ⑪ (index.html) wrapping the `.canvas` element in `<div class = "frame"></div>`
- ⑫ (styles.css) changing the `.frame` border: black solid 50px;
- ⑬ (styles.css) changing the `.frame` width → 500px;
- ⑭ (styles.css) adjust the spacing using padding: 50px;
- ⑮ (styles.css) vertical margin: 20px horizontal margin: auto.  
`margin: 20px auto;` → masih di `.frame` CSS rule.
- ⑯ (index.html) adding new div class `<div class = "one"></div>`
- ⑰ (styles.css) adjust `.one` CSS rule with width: 425px;
- ⑱ (styles.css) height: 150px;
- ⑲ (styles.css) background-color: #efb762.
- ⑳ (styles.css) margin: 20px auto; for `.one` CSS rule.