

Name: Rotha Dapavith

ID: e20190915

Group: I4 GIC(C)

TP07 JavaScript (Node JS, Type Script)

GitHub Repository URL: https://github.com/Dapavith/TP_Internet_Programming/tree/TP07

Ex1: Activity Suggestion

- index.html

```
Exercise01 > index.html > html > body > div.container-box > div#loading.activity
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      <title>Ex1 Activity Suggestion</title>
8      <link rel="stylesheet" href="./style.css">
9  </head>
10 <body>
11     <div class="container-box">
12         <br>
13         <h3>Are you boring?</h3>
14         <button id="button" onclick="loadActivity()">Tell me what to do?</button>
15         <div id="loading" class="activity">
16             <div id="toLoad" class="loading">loading ....</div>
17             <div id="load" class="load">
18                 <h5 id="act" ></h5>
19                 <div class="title">
20                     <p id="type" >Type: </p>
21                     <p id="participant" >Participants: </p>
22                     <p id="price" >Price: $</p>
23                     <p id="link" >Link: https://www.pagat.com</p>
24                 </div>
25             </div>
26         </div>
27     </div>
28
29     <script src="./app.js"></script>
30 </body>
31 </html>
```

-style.css

```
Exercise01 > style.css > ...
1  @import url('https://fonts.googleapis.com');
2  @import url('https://fonts.gstatic.com');
3  @import url('https://fonts.googleapis.com/css2?family=Irish+Grover&display=swap');
4  @import url('https://cdn.jsdelivr.net/npm/font-awesome@6.3.0/css/all.min.css');
5
6  * {
7      margin: 0px;
8      padding: 0px;
9      box-sizing: border-box;
10 }
11
12 body {
13     display: flex;
14     height: 700px;
15     width: auto;
16     justify-content: center;
17     align-items: center;
18 }
19 .container-box {
20     border: 5px solid rgb(32, 31, 31);
21     width: 700px;
22     height: 500px;
23     text-align: center;
24     display: flex;
25     flex-direction: column;
26     align-items: center;
27 }
28
29 .container-box img {
30     width: 190px;
31 }
32
33 .container-box h3 {
34     font-family: 'Irish Grover', cursive;
35 }
36
37 #button {
38     background-color: rgb(10, 110, 280);
39     color: azure;
40     width: 350px;
41     padding: 15px;
42     border: none;
43     border-radius: 15px;
44     font-family: 'Irish Grover', cursive;
45 }
46
47 /* Absolute Center Spinner */
48 .loading {
49     position: fixed;
50     z-index: 999;
51     overflow: show;
52     margin: auto;
53     top: 0;
54     left: 0;
55     bottom: 0;
56     right: 0;
57     width: 50px;
58     height: 50px;
59 }
60
61 /* Transparent Overlay */
62 .loading:before {
63     content: '';
64     display: block;
65     position: fixed;
66     top: 0;
67     left: 0;
68     width: 100%;
69     height: 100%;
70     background-color: rgba(255,255,255,0.5);
```

```

70     background-color: rgba(255,255,255,0.5);
71 }
72
73 /* :not(:required) hides these rules from IE9 and below */
74 .loading:not(:required) {
75     /* hide "loading..." text */
76     font: 0/0 a;
77     color: transparent;
78     text-shadow: none;
79     background-color: transparent;
80     border: 0;
81 }
82
83 .loading:not(:required):after {
84     content: '';
85     display: block;
86     font-size: 10px;
87     width: 50px;
88     height: 50px;
89     margin-top: -0.5em;
90     border: 12px solid rgba(33, 150, 243, 1.0);
91     border-radius: 100%;
92     border-bottom-color: transparent;
93     -webkit-animation: spinner 1s linear 0s infinite;
94     animation: spinner 1s linear 0s infinite;
95 }
96

```

Exercise01 > style.css > ...

```


97     /* Animation */
98
99 @-webkit-keyframes spinner {
100     0% {
101         -webkit-transform: rotate(0deg);
102         -moz-transform: rotate(0deg);
103         -ms-transform: rotate(0deg);
104         -o-transform: rotate(0deg);
105         transform: rotate(0deg);
106     }
107     100% {
108         -webkit-transform: rotate(360deg);
109         -moz-transform: rotate(360deg);
110         -ms-transform: rotate(360deg);
111         -o-transform: rotate(360deg);
112         transform: rotate(360deg);
113     }
114 }
115 @-moz-keyframes spinner {
116     0% {
117         -webkit-transform: rotate(0deg);
118         -moz-transform: rotate(0deg);
119         -ms-transform: rotate(0deg);
120         -o-transform: rotate(0deg);
121         transform: rotate(0deg);
122     }
123     100% {
124         -webkit-transform: rotate(360deg);
125         -moz-transform: rotate(360deg);
126         -ms-transform: rotate(360deg);
127         -o-transform: rotate(360deg);
128         transform: rotate(360deg);
129     }
130 }

```





```

131 @-o-keyframes spinner {
132     0% {
133         -webkit-transform: rotate(0deg);
134         -moz-transform: rotate(0deg);
135         -ms-transform: rotate(0deg);
136         -o-transform: rotate(0deg);
137         transform: rotate(0deg);
138     }
139     100% {
140         -webkit-transform: rotate(360deg);
141         -moz-transform: rotate(360deg);
142         -ms-transform: rotate(360deg);
143         -o-transform: rotate(360deg);
144         transform: rotate(360deg);
145     }
146 }
147 @keyframes spinner {
148     0% {
149         -webkit-transform: rotate(0deg);
150         -moz-transform: rotate(0deg);
151         -ms-transform: rotate(0deg);
152         -o-transform: rotate(0deg);
153         transform: rotate(0deg);
154     }
155     100% {
156         -webkit-transform: rotate(360deg);
157         -moz-transform: rotate(360deg);
158         -ms-transform: rotate(360deg);
159         -o-transform: rotate(360deg);
160         transform: rotate(360deg);
161     }
162 }

```



Exercise01 >  style.css > ...

```

164 .activity {
165     margin: 30px 0px;
166     width: 600px;
167     height: 300px;
168     overflow-x: auto;
169     border: 3px solid  rgb(205, 199, 199);
170     background-color:  rgb(216, 223, 231);
171     border-radius: 10px;
172 }
173
174 .load .title .edit i{
175     width: 25px;
176     height: 25px;
177     color:  rgb(13, 108, 241);
178     margin-top: 6px;
179     margin-right: 12px;
180 }
181
182 .title {
183     padding-left: 70px;
184     text-align: center;
185     font-size: medium;
186 }
187
188 .edit {
189     display:  flex;
190     align-items: center;
191     padding: 17px;
192     height: 25px;
193 }
194
195

```

- app.js

Exercise01 >  app.js >  loadActivity

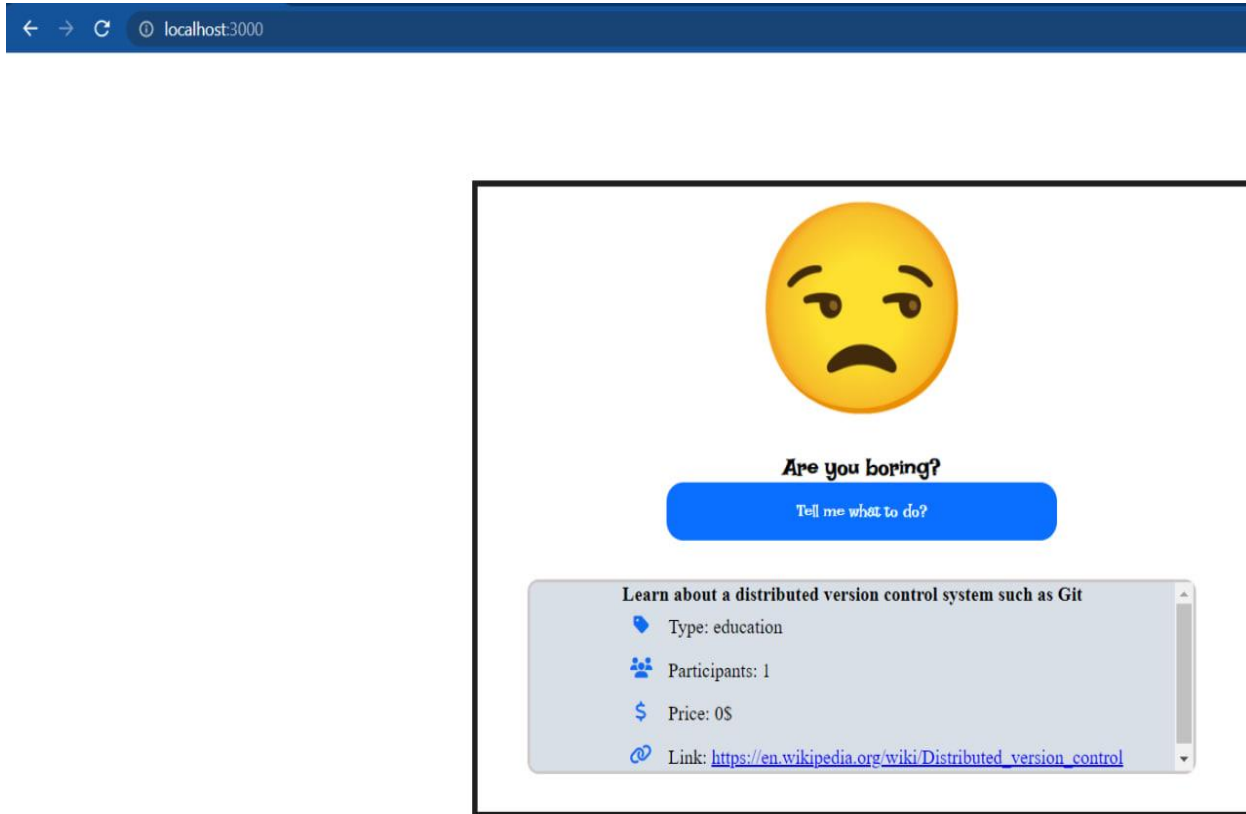
```
1
2 let loading = document.getElementById('loading');
3 loading.hidden = true;
4 function loadActivity(){
5     loading.hidden = false;
6     let toLoad = document.getElementById('toLoad');
7     let load = document.getElementById('load');
8     let display = document.getElementById('display');
9     toLoad.hidden = false;
10    load.hidden = true;
11    fetch('https://www.boredapi.com/api/activity')
12    .then(async (response) =>{
13        toLoad.hidden = true;
14        load.hidden = false;
15        let data = await response.json();
16        load.innerHTML = `
17        <h4>${data["activity"]}</h4>
18        <div class="title">
19            <div class="edit">
20                <i class="fa-solid fa-tag"></i>
21                <p>Type: ${data['type']}</p>
22            </div>
23            <div class="edit">
24                <i class="fa-solid fa-users"></i>
25                <p>Participants: ${data['participants']}</p>
26            </div>
27            <div class="edit">
28                <i class="fa-solid fa-dollar-sign"></i>
29                <p>Price: ${data['price']}</p>
30            </div>
31            <div class="edit">
32                <i class="fa-solid fa-link"></i>
33                <p>Link: <a href="${data['link']}">${data['link']}</a></p>
34            </div>
35            </div>
36        `;
37    });
38 }
```

- index.js

Exercise01 >  index.js > ...

```
1
2 const express = require('express')
3 const path = require('path');
4
5 const app = express();
6
7 app.use(express.static(path.join(__dirname+"/src")));
8
9 app.listen(3000, () => {
10     console.log("The server is running on port 3000");
11 });
```

- Result for Ex1:
- install express: `npm install express`
- start sever: `nodemon npm run start`
- Open localhost: 3000



Exercise02: List and Viewer

`https://api.punkapi.com/v2/beers?`

Page=\${page}&per page=\${limit}`

Build a paginated scrollable list and design a details viewer next to the list.

*** Infinite Scroll + Local store sync

-index.html

```
Exercise03 > index.html > ...
1  <!DOCTYPE html>
2  <html lang="en">
3
4  <head>
5      <meta charset="UTF-8">
6      <meta http-equiv="X-UA-Compatible" content="IE=edge">
7      <meta name="viewport" content="width=device-width, initial-scale=1.0">
8      <link rel="stylesheet" href="./style.css">
9      <title>Exercise 3 List and View</title>
10 </head>
11 <body>
12 <div id="container">
13     <div><h1>Find your Favorite ?</h1></div>
14     <div id="block">
15         <div id="block_list"></div>
16         <div id="block_display"></div>
17     </div>
18 </div>
19 <script src="./listview.js"></script>
20 </body>
21 </html>
22
```

- style.scss

```
Exercise03 > style.scss > .builder
1  *{
2      margin: 0;
3      padding: 0;
4      box-sizing: border-box;
5  }
6  body{
7      display: flex;
8      justify-content: center;
9      margin: 5rem;
10 }
11 #container{
12     width: 1500px;
13     height: 800px;
14     border: 4px solid black;
15     display: grid;
16     justify-content: center;
17 }
18 #container h1{
19     font-family: 'Irish Grover', cursive;
20     padding-top: 50px;
21     text-align: center;
22 }
23 #block{
24     display: flex;
25     justify-content: center;
26 }
27 #block_list{
28     font-family: Arial, Helvetica, sans-serif;
29     width: 400px;
30     height: 500px;
31     background-color: #f4f4f4;
32     margin: 5px;
33     border-radius: 10px;
34     overflow-y: auto;
35 }
```

```

36  #block_display{
37      font-family: Arial, Helvetica, sans-serif;
38      border: 2px solid #f4f4f4;
39      width: 400px;
40      height: 500px;
41      background-color: white;
42      margin: 5px;
43      border-radius: 10px;
44      overflow-y: auto;
45  }
46
47  #loading{
48      margin-top: 250px;
49      width: 50%;
50      height: 100px;
51  }
52  .box_list{
53      display: flex;
54      margin-top: 5px;
55      margin-left: 10px;
56      width: 95%;
57      height: 100px;
58      background-color: white;
59      border-radius: 5px;
60  }
61  .image_list{
62      font-size: 40px;
63  }
64  .box_list:hover{
65      background-color: rgb(64, 105, 210);
66  }
67  .text1{
68      margin-top: 10px;
69  }
70  .text2{
71      margin-top: 10px;
72  }
73  .block_display_beer{
74      display: grid;
75      justify-content: center;
76  }
77  .image_beer{
78      display: flex;
79      justify-content: center;
80  }
81  .beer{
82      width: 200px;
83      height: 400px;
84      margin-top: 20px;
85  }
86  .description{
87      text-align: center;
88  }
89  .name{
90      margin-top: 10px;
91  }
92  .time{
93      margin-top: 10px;
94  }
95  .contribution{
96      margin-top: 10px;
97  }
98  .builder{
99      font-weight: bold;
100 }

```


- Listview.js

```
Exercise03 > 0: listview.js > ...
1  let list_data = document.getElementById("block_list");
2  let dataLoading =false;
3  let page=0;
4  let size=10;
5  let check=1;
6  let tmp="";
7  let parseddata="";
8  function click_display(){
9      let tmp1="";
10     for(let i=0,len=list_data.children.length;i<len;i++){
11         (function(index){
12             list_data.children[i].onclick=function(){
13                 tmp1=`
14                 <div class="block_display_beer">
15                     <div class="image_beer"></img></div>
16                     <div class="description">
17                         <h3 class="name">${parseddata[index].name}</h3>
18                         <p class="time">${parseddata[index].first_brewed}</p>
19                         <p class="des">${parseddata[index].description}</p>
20                         <p class="contribution">Contribution</p>
21                         <p class="bulider"> ( ${parseddata[index].contributed_by} )</p>
22                     </div>
23                 </div>`
24                 document.getElementById("block_display").innerHTML=tmp1;
25             }
26         })(i)
27     }
28 }
29 function display(){
30     let tmp="";
31     const str = localStorage.getItem('data');
32     parseddata = JSON.parse(str);
33     if(str!= null && check ==1){
34         console.log();
35         for(let i=0 ; i<parseddata.length;i++){
36             tmp +=
37             `<div class="box_list">
38                 <div class="image_list"> </div>
39                 <div>
40                     <p class="text1">${parseddata[i].name}</p>
41                     <p class="text2">${parseddata[i].description.substring(0, 30).concat('...')}</p>
42                 </div>
43             </div>`
44         }
45         list_data.innerHTML=tmp;
46         number_page=parseddata.length/10;
47         page=parseddata.length;
48         size=page+10;
49     }
50     if(size<81){
51         fetch(`https://api.punkapi.com/v2/beers? page=${page}&per_page=${size}`).then(async (res) =>{
52             let data_list = await res.json();
53             const data_json = JSON.stringify(data_list);
54             window.localStorage.setItem('data', data_json);
55             for(let i=page ; i<data_list.length;i++){
56                 tmp +=
57                 `<div class="box_list">
58                     <div> </div>
59                     <div class="list_each_data">
60                         <p class="text1">${data_list[i].name}</p>
61                         <p class="text2">${data_list[i].description.substring(0, 30).concat('...')}</p>
62                     </div>
63                 </div> `
64             }
65             document.getElementById("block_list").innerHTML=tmp;
66             click_display()
67         })
68     }
69 }
```

```

70 list_data.addEventListener("scroll", (event) => {
71     const { scrollTop, clientHeight, scrollHeight } = list_data;
72     if ((clientHeight + scrollTop) >= scrollHeight*0.98) {
73         if (!dataLoading) {
74             size = page +10;
75             display()
76         }
77     }
78 }
79 });
80 display();
81 click_display();
82

```

- index2.js

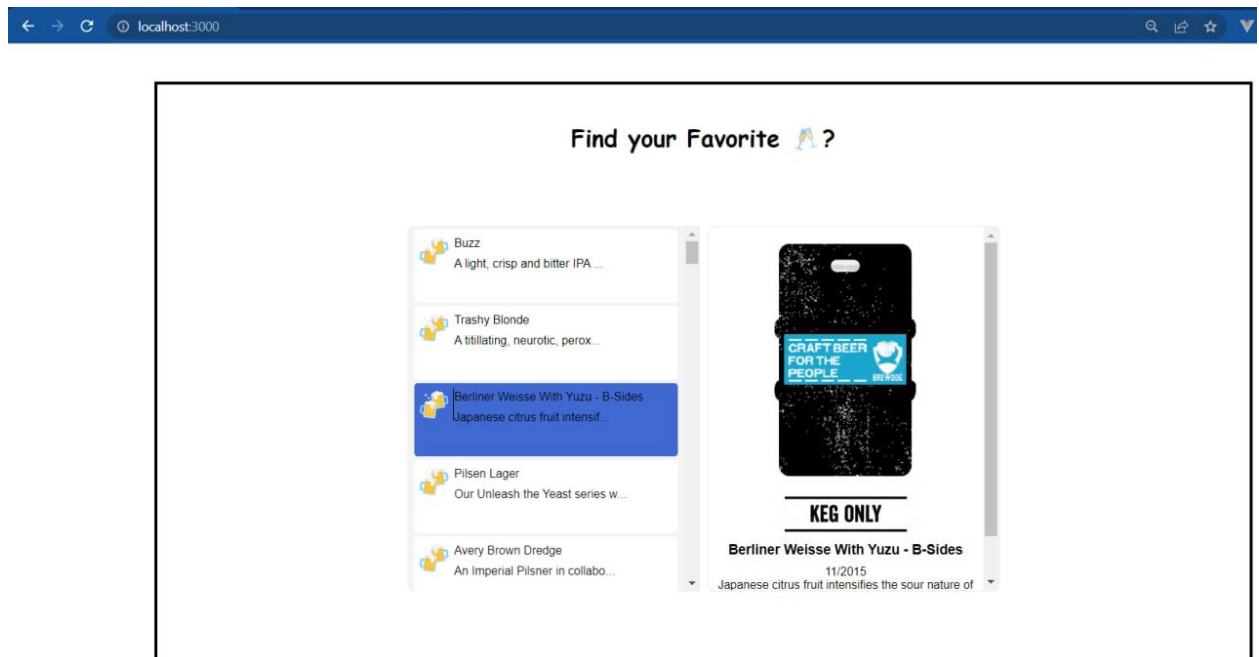
```

Exercise02 > JS index2.js > ...
1
2 const express = require('express')
3 const path = require('path');
4
5 const app = express();
6
7 app.use(express.static(path.join(__dirname+"/src")));
8
9 app.listen(3000, () => {
10     console.log("The server is running on port 3000");
11 });

```

- Result for Ex02:

- install express: npm install express
- start sever: nodemon npm run start
- Open localhost: 3000



Fetch Data from API for book App

Ex03: Book Store

- index.html

```
Exercise05 > index.html > html > body
1  <!DOCTYPE html>
2  <html lang="en">
3
4  <head>
5    <meta charset="UTF-8">
6    <meta http-equiv="X-UA-Compatible" content="IE=edge">
7    <meta name="viewport" content="width=device-width, initial-scale=1.0">
8    <link rel="shortcut icon" href="./img/book.png" type="image/x-icon">
9    <title>Ex05 Book Shop Crud</title>
10   <link rel="stylesheet" href="./style.css">
11 </head>
12
13 <body>
14   <div class="loading">
15     <div class="loader"></div>
16   </div>
17   <div class="container">
18     <form autocomplete="off" onsubmit="onFormSubmit()">
19       <div class="add">
20         <div class="header-add">Add new book here!</div>
21
22         <label for="name_book">Name</label>
23         <input type="text" name="name" id="name_book" placeholder="c programming" onclick="if(this.placeholder == 'c programming') {this.placeholder = ''}"
24         onBlur="if(this.placeholder == '') {this.placeholder = 'c programming'}" />
25         <label for="category_book">Category</label>
26         <input type="text" name="category" id="category_book" placeholder="coding" onclick="if(this.placeholder == 'coding') {this.placeholder = ''}"
27         onBlur="if(this.placeholder == '') {this.placeholder = 'coding'}" />
28         <label for="price_book">Price</label>
29         <input type="text" name="price_book" id="price_book" placeholder="1000 riels" onclick="if(this.placeholder == '1000 riels') {this.placeholder = ''}"
30         onBlur="if(this.placeholder == '') {this.placeholder = '1000 riels'}" />
31
32         <button type="submit" class="add_book">Add</button>
33         <button type="submit" class="edit_book">update</button>
34
35       </div>
36     </form>
37     <div class="home" id="add_items">
38       <div class="book-info">List Book</div>
39     </div>
40
41     <script src="./book.js"></script>
42   </div>
43 </body>
44 </html>
```

- fetchapi-detail.html

```
Exercise05 > fetchapi-detail.html > html > head > link
1  <!DOCTYPE html>
2  <html lang="en">
3
4  <head>
5      <meta charset="UTF-8">
6      <meta http-equiv="X-UA-Compatible" content="IE=edge">
7      <meta name="viewport" content="width=device-width, initial-scale=1.0">
8      <title>Book Details</title>
9      <link rel="stylesheet" href="./api-book.css">
10 </head>
11
12 <body>
13     <div class="items"></div>
14     <script src="./book-api.js"></script>
15 </body>
16
17 </html>
```

- style.css

```
Exercise05 > style.css > .add>div
1  * {
2      margin: 0px;
3      padding: 0px;
4      box-sizing: border-box;
5  }
6
7  body {
8      background-color: #aaa;
9  }
10
11  .container {
12      display: flex;
13      justify-content: center;
14  }
15
16  .add {
17      height: 93vh;
18      width: 300px;
19      background-color: antiquewhite;
20      position: fixed;
21      padding: 15px;
22      border-radius: 10px;
23  }
24
25  .add>div {
26      text-align: center;
27      color: #ffffff;
28      background-color: rgb(147, 212, 116);
29      margin-bottom: 20px;
30      font-size: large;
31      padding-top: 10px;
32      height: 30px;
33      border-radius: 5px;
34  }
35
36  .add input[type=text] {
37      width: 100%;
38      border: 2px solid #aaa;
39      border-radius: 4px;
40      margin: 8px 0;
41      outline: none;
42      padding: 8px;
43      box-sizing: border-box;
44      transition: .3s;
45  }
46
```

Exercise05 > style.css > .add>div

```
44     transition: .3s;
45 }
46
47 ✓ .add input[type=text]:focus {
48     border-color: dodgerBlue;
49     box-shadow: 0 0 8px 0 dodgerBlue;
50 }
51
52 ✓ .add .edit_book {
53     visibility: hidden;
54 }
55
56 .add_book,
57 ✓ .edit_book {
58     background-color: rgba(147, 196, 125, 1);
59     color: white;
60     border-style: outset;
61     border-color: #0066A2;
62     height: 25px;
63     width: 50px;
64     font: bold15px arial, sans-serif;
65     text-shadow: none;
66     height: 30px;
67     margin-top: 10px;
68     border-radius: 5px;
69 }
70
71 ✓ .edit_book {
72     background-color: rgb(61, 133, 198);
73 }
74
75 ✓ .home {
76     height: auto;
77     width: 800px;
78     top: 0px;
79     /* background-color: #fafafa; */
80     /* rgba(97, 105, 122, 0.5) */
81     display: flex;
82     flex-wrap: wrap;
83     margin-left: 350px;
84     border-radius: 10px;
85     background-color: #ffffff;
86 }
87
```

```
87
88  ∨ .home .book-info {
89      background-color: rgba(147, 196, 125, 1);
90      text-align: center;
91      height: 35px;
92      font-size: large;
93      color: #ffffff;
94      position: sticky;
95      width: 780px;
96      margin: 10px 0 10px 10px;
97      border-radius: 5px;
98  }
99
100  ∨ .home .items {
101      text-align: center;
102      width: 23%;
103      margin: 1%;
104      margin-left: 1%;
105      box-shadow: rgba(15, 15, 15, 0.3) 0px 7px 29px 0px;
106      border-radius: 10px;
107  }
108
109  ∨ .items>img {
110      margin-top: 10px;
111      margin-bottom: 10px;
112  }
113
114  .items .name,
115  .price,
116  ∨ .category {
117      text-align: start;
118      margin-left: 10px;
119      padding-top: 10px;
120  }
121
122  ∨ .items>button {
123      text-align: start;
124  }
125
126  ∨ .button {
127      margin-top: 10px;
128      margin-bottom: 5px;
129  }
130
```

```
131 .button .editBtn,
132 .deleteBtn {
133     border-radius: 5px;
134     background-color: rgb(61, 133, 198);
135     color: white;
136     border-style: outset;
137     border-color: #0066A2;
138     text-shadow: none;
139     height: 30px;
140 }
141
142 .deleteBtn {
143     margin-left: 4px;
144     background-color: rgba(239, 54, 54);
145 }
146
147 .edit_book {
148     margin-left: -55px;
149     text-align: center;
150     width: 60px;
151 }
152
153 .items>img {
154     height: 90px;
155 }
156
157 .loading {
158     justify-content: center;
159     align-items: center;
160 }
161
162 .btn {
163     margin: 5px 0 15px 0;
164     background-color: rgba(244, 67, 54, .9);
165 }
166
167 .btn>a {
168     color: white;
169     text-decoration: none;
170 }
171
172 @import url(https://fonts.googleapis.com/css?family=Roboto:100);
173
174 @import url(https://fonts.googleapis.com/css?family=Roboto:100);
175 h1 {
176     font: 2em 'Roboto', sans-serif;
177     margin-bottom: 40px;
178 }
179
180 .loader {
181     display: inline-block;
182     width: 50px;
183     height: 50px;
184     margin: 50px;
185     padding: 50px;
186     border: 3px solid rgba(51, 221, 13, 0.804);
187     border-radius: 50%;
188     border-top-color: #ca1010;
189     -webkit-animation: spin 1s ease-in-out infinite;
190     animation: spin 1s ease-in-out infinite;
191 }
192
193 @keyframes spin {
194     to {
195         -webkit-transform: rotate(360deg);
196     }
197 }
198
199 @-webkit-keyframes spin {
200     to {
201         -webkit-transform: rotate(360deg);
202     }
203 }
```

- api-book.css

```
Exercise05 > api-book.css > .home>img
1      *{
2          margin: 0px;
3          padding: 0px;
4          box-sizing: border-box;
5      }
6
7      body {
8          padding: 10px;
9          margin: 10px;
10     }
11
12     .container {
13         height: 10rem;
14         display: flex;
15         justify-content: center;
16         align-items: center;
17         flex-wrap: wrap;
18         background-color: rgba(88, 90, 84, 0.8);
19     }
20
21     .container>div {
22         width: 100%;
23         text-align: center;
24     }
25
26     .home {
27         margin: 2rem 0 0 0;
28         display: flex;
29         justify-content: center;
30         flex-wrap: wrap;
31     }
32
33     .home>div {
34         margin-bottom: 1rem;
35         width: 100%;
36         text-align: center;
37     }
38
39     .home>img {
40         height: 100px;
41     }
```



```

Exercise05 > bookjs > deleteBook
1  var selectedRow = null;
2      let header = document.getElementsByClassName("header-add")[0];
3
4      //read data from local storage and display
5      let books = [];
6      const getB = localStorage.getItem('books');
7      const dataB = JSON.parse(getB);
8
9      let results = [];
10     let output = '';
11
12     var loading = (value) => {
13         if (value == 'true') {
14             setTimeout(() => {
15
16                 }, 1000)
17             document.getElementsByClassName('loading')[0].style.display = 'flex'
18             document.getElementsByClassName('container')[0].style.display = 'none'
19         } else {
20             document.getElementsByClassName('container')[0].style.display = 'flex';
21             document.getElementsByClassName('loading')[0].style.display = 'none';
22         }
23     }
24     loading("true");
25
26     fetch('https://jsonplaceholder.typicode.com/photos').then(async(res) => {
27         results = await res.json();
28         var title = '';
29         for (result in results) {
30             output +=
31                 `<div class="items">` +
32                 `` +
33                 `<div class="name">Title : ${results[result].title.toString().substring(0, 10)}...</div>` +
34                 `<div class="price">Id : ${results[result].id}</div>` +
35                 `<div class="category">Album ID : ${results[result].albumId}</div>` +
36                 `<button class="btn">` +
37                 `<a href="/fetchapi-detail.html?bookid=${parseInt(result)+1}">see more</a>` +
38                 `</button>` +
39                 `</div>`;
40         }
41         document.getElementById("add_items").innerHTML += output;
42         setTimeout(() => {
43             loading("false");
44         }, 100)
45     });
46
47     function onFormSubmit(e) {
48         e.preventDefault();
49         var formData = readFormData();
50         if (selectedRow === null) {
51             insertNewRecord(formData);
52         } else {
53             updateRecord(formData);
54         }
55         resetForm();
56     }
57
58     //clear placeholder
59     document.querySelector('.add_book').placeholder = '';
60
61     //Retrieve the data
62     function readFormData() {
63         var formData = {};
64         formData["name_book"] = document.getElementById("name_book").value;
65         formData["price_book"] = document.getElementById("price_book").value;
66         formData["category_book"] = document.getElementById("category_book").value;
67         return formData;
68     }
69
70     //Insert the data
71     function insertNewRecord(data) {
72         const getdata = localStorage.getItem('books');
73         let databookss = JSON.parse(getdata);
74
75         books = Object.assign([], databookss);
76         let x;
77         if (databookss == null) {
78             x = 0
79         } else {
80             x = databookss.length;
81         }
82         const book = {
83             name: `${data.name_book}`,
84             price: `${data.price_book}`,
85             category: `${data.category_book}`
86         };
87     };

```

Exercise05 > bookjs > deleteBook

```
87     category: `${data.category_book}`
88   };
89   books[x] = book;
90   localStorage.setItem('books', JSON.stringify(books));
91   let databooks = getbooks();
92
93   document.getElementById("add_items").innerHTML +=
94     `<div class="items">` +
95     `` +
96     `<div class="name">Name : ${databooks[x].name}</div>` +
97     `<div class="price">Price : ${databooks[x].price} riels</div>` +
98     `<div class="category">Category : ${databooks[x].category}</div>` +
99     `<div class="button">` +
100     `<button id="${parseInt(x)}" class="editBtn" onclick="editBook(this.id)">Edit</button>` +
101     `<button id="${parseInt(x)}" class="deleteBtn" onclick="deleteBook(this.id)">Delete</button>` +
102     `</div>` +
103     `</div>`;
104 }
105
106 //Edit the data
107 function editBook(id) {
108   header.innerHTML = "Edit information here!";
109   header.style.backgroundColor = "rgb(61,133,198)";
110
111   let getdatabooks = getbooks();
112
113   selectedRow = id;
114   document.getElementsByClassName("edit_book")[0].style.visibility = 'visible';
115   document.getElementsByClassName("add_book")[0].style.visibility = 'hidden';
116
117   document.getElementById('name_book').value = getdatabooks[id].name;
118   document.getElementById('price_book').value = getdatabooks[id].price;
119   document.getElementById('category_book').value = getdatabooks[id].category;
120 }
121
122 function updateRecord(formData) {
123   document.getElementsByClassName("name")[selectedRow].innerHTML = "Name : " + formData.name_book;
124   document.getElementsByClassName("price")[selectedRow].innerHTML = "Price : " + formData.price_book + " riels";
125   document.getElementsByClassName("category")[selectedRow].innerHTML = "Category : " + formData.category_book;
126   document.getElementsByClassName("edit_book")[0].style.visibility = 'hidden';
127   document.getElementsByClassName("add_book")[0].style.visibility = 'visible';
128 }
```

Exercise05 > bookjs > deleteBook

```
125 document.getElementsByClassName("category")[selectedRow].innerHTML = "Category : " + formData.category_book;
126 document.getElementsByClassName("edit_book")[0].style.visibility = 'hidden';
127 document.getElementsByClassName("add_book")[0].style.visibility = 'visible';
128
129 //get books from localStorage
130 let bookss = getbooks();
131 const bookUpdate = {
132   name: `${formData.name_book}`,
133   price: `${formData.price_book}`,
134   category: `${formData.category_book}`
135 };
136
137 //merge bookss to books
138 books = Object.assign([], bookss);
139 books[selectedRow] = bookUpdate;
140 localStorage.setItem('books', JSON.stringify(books));
141
142 document.getElementsByClassName("header-add")[0].innerHTML = "Add new book here!";
143 header.style.backgroundColor = "rgb(147, 196, 125)";
144 selectedRow = null;
145 }
146
147 //delete
148 function deleteBook(id) {
149   //get books from localStorage
150   let databooks = getbooks();
151   if (confirm('Do you want to delete this record?')) {
152     //remove books at id and count 1
153     databooks.splice(id, 1);
154     localStorage.setItem('books', JSON.stringify(databooks));
155
156     //reset home to update id element
157     document.getElementById("add_items").innerHTML = '';
158     document.getElementById("add_items").innerHTML = `<div class="book-info">Book</div>`;
159     for (databook in databooks) {
160       document.getElementById("add_items").innerHTML +=
161         `<div class="items">` +
162         `` +
163         `<div class="name">Name : ${databooks[databook].name}</div>` +
164         `<div class="price">Price : ${databooks[databook].price} riels</div>` +
165         `<div class="category">Category : ${databooks[databook].category}</div>` +
166         `<div class="button">` +
167         `<button id="${parseInt(databook)}" class="editBtn" onclick="editBook(this.id)">Edit</button>` +
168         `<button id="${parseInt(databook)}" class="deleteBtn" onclick="deleteBook(this.id)">Delete</button>` +
169         `</div>` +
170         `</div>`;
```

```

Exercise05 > bookjs > deleteBook
168         <button id="${parseInt(databook)}" class="deleteBtn" onclick="deleteBook(this.id)">Delete</button>` +
169         `</div>` +
170         `</div>`;
171     }
172 }
173 resetForm();
174 }
175
176 function resetForm() {
177     document.getElementById('name_book').value = '';
178     document.getElementById('price_book').value = '';
179     document.getElementById('category_book').value = '';
180 }
181
182 //get books from localStorage
183 function getbooks() {
184     const getbooks = localStorage.getItem('books');
185     let databooks = JSON.parse(getbooks);
186
187     return databooks;
188 }

```

- book-api.js

```

Exercise05 > JS book-api.js > ...
1  const queryString = window.location.search;
2      const urlParams = new URLSearchParams(queryString);
3
4      const id = urlParams.get('bookid');
5      let result = [];
6      let output = '';
7
8      fetch(`https://jsonplaceholder.typicode.com/photos/${id}`).then(async(res) => {
9          result = await res.json();
10         console.log(result)
11         output += `<div class="container">` +
12             `<div>${result.title}</div>` +
13             `<div>${result.albumId}</div>` +
14             `</div>` +
15             `<div class="home">` +
16             `<div>${result.id}</div>` +
17             `` +
18             `</div>`
19
20         document.getElementsByClassName('items')[0].innerHTML = output;
21     });

```

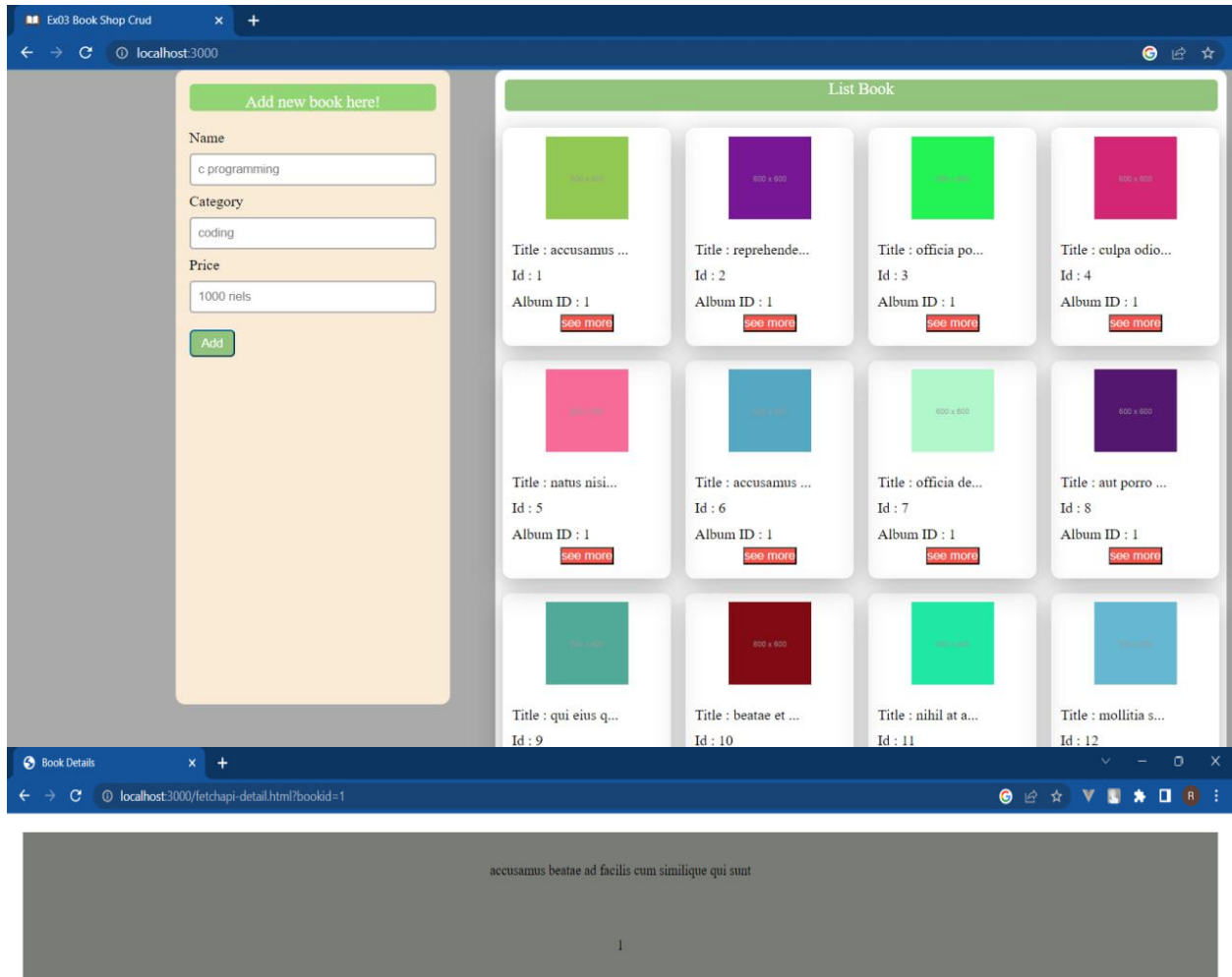
- index.js

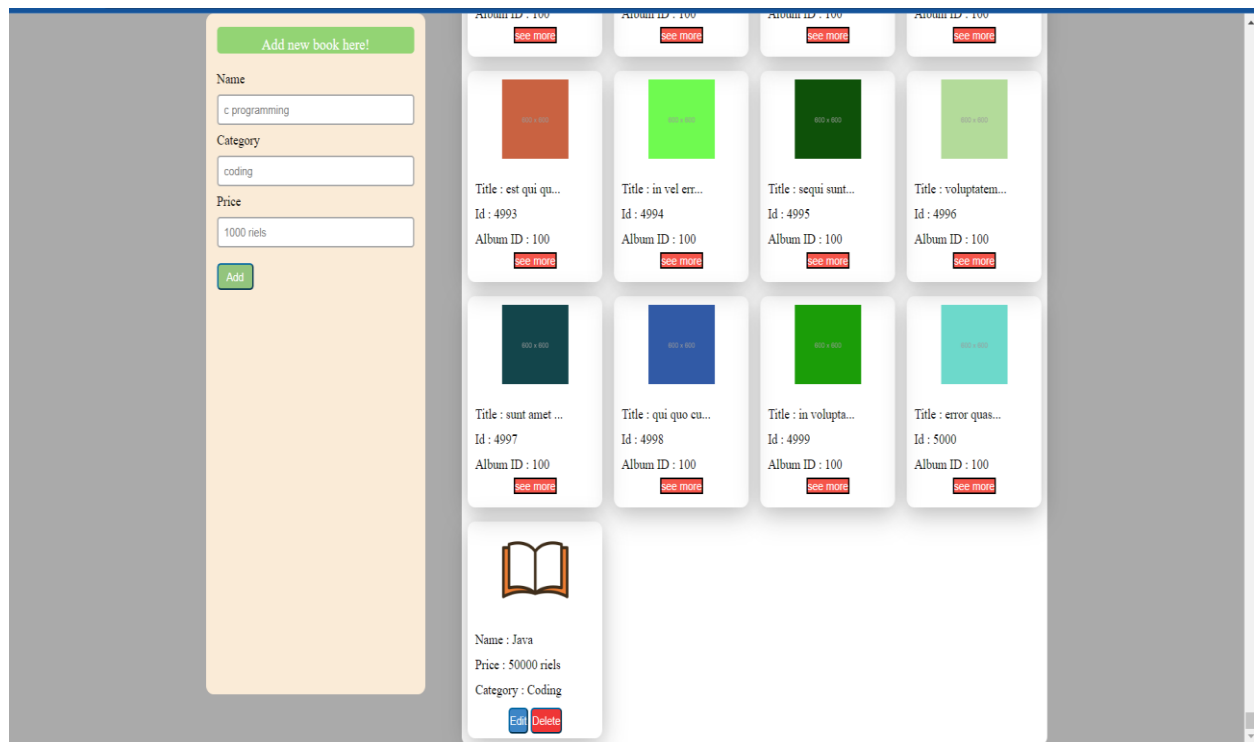
```

Exercise03 > JS index.js > ...
1  const express = require('express');
2  const path = require("path");
3  const app = express();
4  const fs = require('fs');
5
6
7  app.get('/detail', function (req, res) {
8
9      fs.readFile('./src/fetchapi-detail.html', null, (err, data) => {
10         res.write(data)
11     })
12 });
13
14 app.use(express.static(path.join(__dirname+"/src")));
15 app.listen(3000, () => {
16     console.log("The Server is running on port: 3000");
17 });

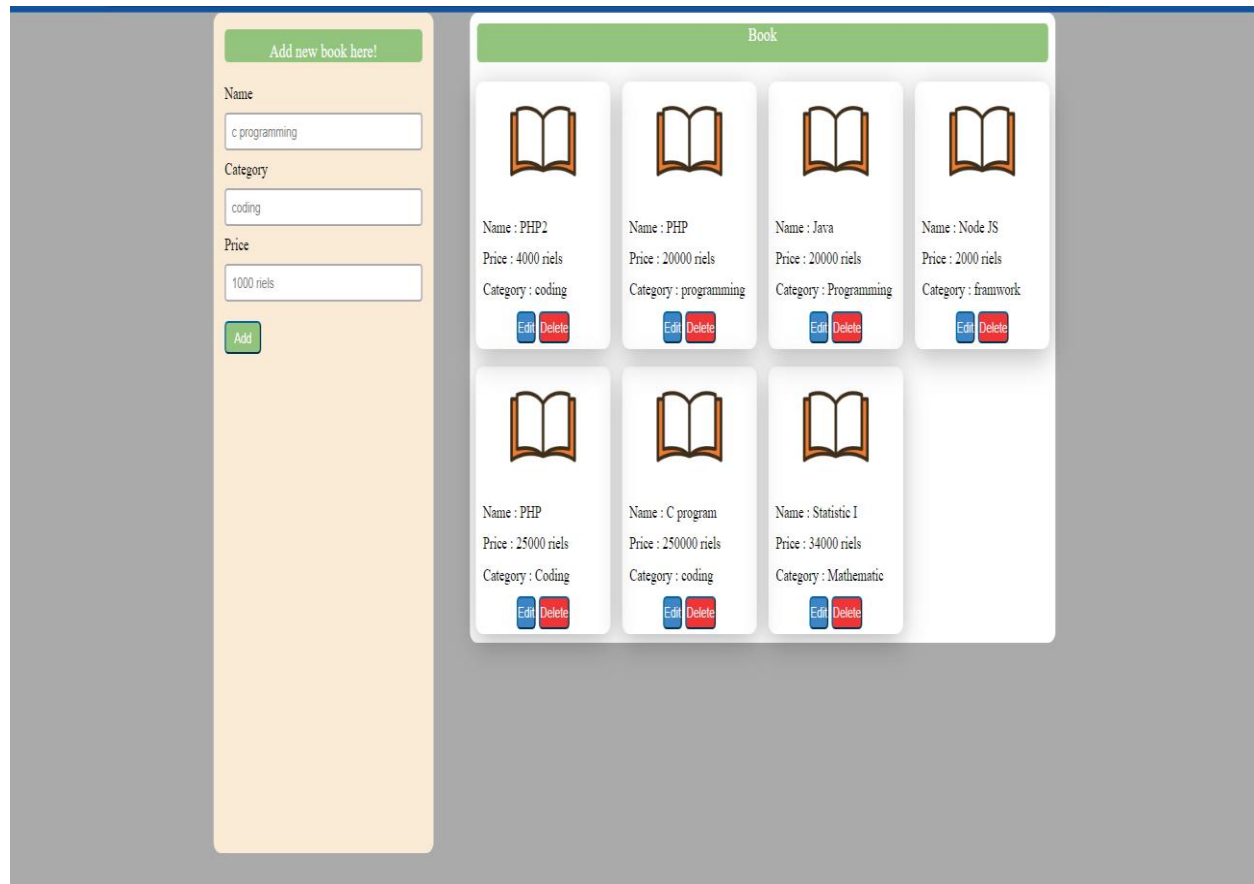
```

- Result of Ex3:
- install express: npm install express
- start sever: nodemon npm run start
- Open localhost: 3000





- After delete we will see record data of list book



Ex04:

Past Time Library with Typescript Write a Typescript library for Khmer DateTime. The library can be imported to use in JavaScript module.

- index.html


```
Exercise04 > src > index.html > "index.html" > html > body > script > onchange
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   <title>Ex4 Kh Past Time Library with TS</title>
8   <script src="https://cdn.tailwindcss.com"></script>
9 </head>
10
11 <body>
12   <div class="wrapper flex flex-col w-full h-screen justify-center items-center">
13     
14     <div class="font-serif font-bold text-3xl mt-5">Khmer Past Time</div>
15     <input type="datetime-local" id="valDate" class="border bg-gray-200 rounded-lg pl-3 pr-3 p-1 mt-5 mb-10">
16     <div id="status" class="text-4xl font-bold text-orange-700"></div>
17   </div>
18   <script type="module">
19     import { KhmerDate } from "./assets/dateChecker.js";
20
21     var date_input = document.getElementById('valDate');
22
23     date_input.onchange = function () {
24
25       const in_date = new Date(this.value);
26       const inputDate = new KhmerDate(new Date(in_date.getTime()));
27
28       console.log(inputDate.getRelativeTime());
29       console.log(inputDate.getDate());
30       console.log(Date.now());
31
32       document.getElementById('status').innerHTML = inputDate.getRelativeTime();
33       console.log(this.value);
34     }
35   </script>
36 </body>
37 </html>
```

- Time.html




```
Exercise04 > src > ex4 > Time.html > "Time.html" > html > body > div.wrapper.flex.flex-col.w-full.h-screen.justify-center.items-center.bg-gray-200 > img.w-[12rem]
1 <!DOCTYPE html>
2 <html lang="en">
3
4 <head>
5   <meta charset="UTF-8">
6   <meta http-equiv="X-UA-Compatible" content="IE=edge">
7   <meta name="viewport" content="width=device-width, initial-scale=1.0">
8   <title>Past Time</title>
9   <script src="https://cdn.tailwindcss.com"></script>
10 </head>
11 <body>
12   <div class="wrapper flex flex-col w-full h-screen justify-center items-center bg-gray-200">
13     
14     <div class="font-serif font-bold text-3xl mt-5">Khmer Past Time</div>
15     <input type="datetime-local" id="valDate" class="border bg-gray-500 rounded-lg pl-3 pr-3 p-1 mt-5 mb-10">
16     <div id="status" class="text-4xl font-bold text-orange-600"></div>
17   </div>
18   <script type="module">
19     import { KhmerDate } from "./assets/dateChecker.js";
20     var date_input = document.getElementById('valDate');
21     date_input.onchange = function () {
22       const in_date = new Date(this.value);
23       const inputDate = new KhmerDate(new Date(in_date.getTime()));
24       console.log(inputDate.getRelativeTime());
25       console.log(inputDate.getDate());
26       console.log(Date.now());
27       document.getElementById('status').innerHTML = inputDate.getRelativeTime();
28       console.log(this.value);
29     }
30   </script>
31 </body>
32 </html>
```

- assets



- khmerNumConverter.js

```
Exercise04 > src > assets >  khmerNumConverter.js > ...
1  function convertToKhmerNumber(number) {
2      const khmerNums = ["០", "១", "២", "៣", "៤", "៥", "៦", "៧", "៨", "៩"];
3      let result = "";
4      const numString = number.toString();
5      for (let i = 0; i < numString.length; i++) {
6          const digit = parseInt(numString.charAt(i));
7          result += khmerNums[digit];
8      }
9      return result;
10 }
11
12 export { convertToKhmerNumber };
```

- dateChecker.js

```
Exercise04 > src > assets >  dateChecker.js >  KhmerDate >  getRelativeTime
1  import { convertToKhmerNumber } from "../khmerNumConverter.js";
2  class KhmerDate {
3      constructor(date) {
4          this.date = date;
5      }
6      getDate() {
7          return this.date;
8      }
9      getRelativeTime() {
10         const now = new Date();
11         const msAgo = now.getTime() - this.date.getTime();
12         if (msAgo < 60 * 1000) {
13             return `មុននេះបន្តិច`;
14         }
15         else if (msAgo < 60 * 60 * 1000) {
16             return `${convertToKhmerNumber(Math.floor(msAgo / (60 * 1000)))} នាទីមុន`;
17         }
18         else if (msAgo < 24 * 60 * 60 * 1000) {
19             return `${convertToKhmerNumber(Math.floor(msAgo / (60 * 60 * 1000)))} ម៉ោងមុន`;
20         }
21         else if (msAgo < 7 * 24 * 60 * 60 * 1000) {
22             return `${convertToKhmerNumber(Math.floor(msAgo / (24 * 60 * 60 * 1000)))} ថ្ងៃមុន`;
23         }
24         else if (msAgo < 30 * 24 * 60 * 60 * 1000) {
25             return `${convertToKhmerNumber(Math.floor(msAgo / (7 * 24 * 60 * 60 * 1000)))} សប្តាហ៍មុន`;
26         }
27         else if (msAgo < 365 * 24 * 60 * 60 * 1000) {
28             return `${convertToKhmerNumber(Math.floor(msAgo / (30 * 24 * 60 * 60 * 1000)))} ខែមុន`;
29         }
30         else {
31             return `${convertToKhmerNumber(Math.floor(msAgo / (365 * 24 * 60 * 60 * 1000)))} ឆ្នាំមុន`;
32         }
33     }
34 }
35 export { KhmerDate };
36
```

- dateChecker.ts

```
Exercise04 > src > assets >  dateChecker.ts >  KhmerDate
1  export class KhmerDate {
2      private readonly date: Date;
3      constructor(date: Date) {
4          this.date = date;
5      }
6      getDate(): Date {
7          return this.date;
8      }
9      getRelativeTime(): string {
10         const now = new Date();
11         const msAgo = now.getTime() - this.date.getTime();
12         if (msAgo < 60 * 1000) {
13             return `មុននេះបន្តិច`;
14         }
15         else if (msAgo < 60 * 60 * 1000) {
16             return `${Math.floor(msAgo / (60 * 1000))} នាទីមុន`;
17         }
18         else if (msAgo < 24 * 60 * 60 * 1000) {
19             return `${Math.floor(msAgo / (60 * 60 * 1000))} ម៉ោងមុន`;
20         }
21         else if (msAgo < 7 * 24 * 60 * 60 * 1000) {
22             return `${Math.floor(msAgo / (24 * 60 * 60 * 1000))} ថ្ងៃមុន`;
23         }
24         else if (msAgo < 30 * 24 * 60 * 60 * 1000) {
25             return `${Math.floor(msAgo / (7 * 24 * 60 * 60 * 1000))} សប្តាហ៍មុន`;
26         }
27         else if (msAgo < 365 * 24 * 60 * 60 * 1000) {
28             return `${Math.floor(msAgo / (30 * 24 * 60 * 60 * 1000))} ខែមុន`;
29         }
30         else {
31             return `${Math.floor(msAgo / (365 * 24 * 60 * 60 * 1000))} ឆ្នាំមុន`;
32         }
33     }
34 }
```


- index.js

```
Exercise04 >  index.js > ...
1  const express = require('express');
2  const fs = require('fs');
3  const app = express();
4  app.use(express.static('src/'));
5
6  app.get('/', (req, res) => {
7      fs.readFile('./src/index.html', 'utf8', (err, data) => {
8          if (err) {
9              //If there's an error send request status
10             res.status(500).send('Error reading file');
11         } else {
12             res.send(data);
13         }
14     });
15 });
16
17 app.listen(3000, () => {
18     console.log('Server listening on port 3000');
19 });|
```


- Result for Ex04:
- install express: `npm install express`
- install typescript: `npm i -g typescript`
- start sever: `nodemon npm run start`
- Open localhost: 3000



Khmer Past Time

03/14/2023 06:24 PM 

មុននេះបន្តិច