

## 3SA03 : React

### งานหลังการทดลอง

จากการศึกษาเพิ่มเติมเกี่ยวกับ React และปรับเปลี่ยนเกม Logic หลังทำการทดลองได้ผลลัพธ์ดังนี้

#### 1. เปลี่ยนสีตัวอักษร โดยสามารถเปลี่ยนอักษรสีแดงเป็นสีเขียว

**เปลี่ยนสีตัวอักษร**

👤 master

🌿 Donlee152 committed 1 hour ago

Showing 1 changed file with 1 addition and 1 deletion.

```

src/App.css
...  ... @@ -1,14 +1,14 @@
1 1  .card {
2 2    display: inline-block;
3 3    text-align: center;
4 4    width: 3em;
5 5    font-size: 2em;
6 6    box-shadow: 0 4px 8px 0 rgba(0,0,0,0.2);
7 7    margin: 1em;
8 8    user-select: none;
9 9  }
10 10
11 11  .activeCard{
12 -   color: red;
12 +   color: green;
13   background: pink;
14 14  }

```

#### 2. เปลี่ยนขนาดกล่องใส่ตัวอักษร

```

src/App.css
...  ... @@ -1,8 +1,8 @@
1 1  .card {
2 2    display: inline-block;
3 3    text-align: center;
4 -   width: 3em;
5 -   font-size: 2em;
4 +   width: 6em;
5 +   font-size: 5em;
6 6    box-shadow: 0 4px 8px 0 rgba(0,0,0,0.2);
7 7    margin: 1em;
8 8    user-select: none;
9 9  }
10 10
11 11  .activeCard{
12 12    color: blue;
13 13    background: pink;
14 14  }

```

### 3. เปลี่ยนรูปแบบของ display

```

src/App.css
... -1,8 +1,31 @@
1 + .App {
2 +   text-align: center;
3 + }
4 + .App-logo {
5 +   animation: App-logo-spin infinite 20s linear;
6 +   height: 80px;
7 + }
8 + .App-header {
9 +   background-color: #222;
10 +   height: 150px;
11 +   padding: 20px;
12 +   color: white;
13 + }
14 + .App-title {
15 +   font-size: 1.5em;
16 + }
17 + .App-intro {
18 +   font-size: large;
19 + }
20 + @keyframes App-logo-spin {
21 +   from { transform: rotate(0deg); }
22 +   to { transform: rotate(360deg); }
23 + }
24 .card {
25 - display: inline-block;
26 + display: unset;
27   text-align: center;
28 - width: 6em;
29 - font-size: 5em;
30 + width: 3em;
31 + font-size: 2em;
32   box-shadow: 0 4px 8px 0 rgba(0,0,0,0.2);
33   margin: 1em;
34   user-select: none;
35 }
36 .activeCard{
37   color: blue;
38   background: pink;
39 }

```

### 4. ทดลองเปลี่ยนคำและอักษรที่แสดงผลผ่านทางหน้าจอ

```

src/App.js
... -1,14 +1,16 @@
1 - import React from 'react';
2 + import React, { Component } from 'react';
3   import './App.css';
4   import WordCard from './WordCard';

```

5	-	const word = "Hello";
6	-	function App() {
5	+	const word = "JavaScript";
6	+	class App extends Component {
7	+	render() {
7	8	return (
8	9	<div>
9	10	<WordCard value={word}/>
10	11	</div>
11	12	);
13	+	}
12	14	}
13	15	
14	16	export default App;

## 5. ทดลองเปลี่ยนสีพื้นหลังของเกมส Logic

...	...	@@ -1,6 +1,9 @@
1	1	.App {
2	2	text-align: center;
3	3	+ background-color: blueviolet;
4	4	+ height: 100vh;
3	5	}
6	6	+
4	7	.App-logo {
5	8	animation: App-logo-spin infinite 20s linear;
6	9	height: 80px;
7	10	}
8	11	.App-header {
9	12	background-color: #222;
10	13	height: 150px;
11	14	padding: 20px;
12	15	color: white;
13	16	}
14	17	.App-title {
15	18	font-size: 1.5em;
16	19	}
17	20	.App-intro {
18	21	font-size: large;
19	22	}
20	23	@keyframes App-logo-spin {
21	24	from { transform: rotate(0deg); }
22	25	to { transform: rotate(360deg); }
23	26	}
24	27	.card {
25	28	display: inline-block;
26	29	text-align: center;
27	30	width: 3em;
28	31	font-size: 2em;
29	32	box-shadow: 0 4px 8px 0 rgba(0,0,0,0.2);
30	33	margin: 1em;
31	34	user-select: none;
32	35	}
33	36	.activeCard {
34	37	color: blue;
35	38	background: pink;
36	39	}

## 6. ทดลองปรับเปลี่ยนสีและขนาดตัวอักษร

```

16  src/App.css
...  ...  @@ -1,9 +1,8 @@
1  1  .App {
2  2  text-align: center;
3  - background-color: blueviolet;
3  + background-color: chartreuse;
4  4  height: 100vh;
5  5  }
6  -
7  6  .App-logo {
8  7  animation: App-logo-spin infinite 20s linear;
9  8  height: 80px;
10  9  }
11  10 .App-header {
12  11 background-color: #222;
13  12 height: 150px;
14  13 padding: 20px;
15  14 color: white;
16  15 }
17  16 .App-title {
18  17 font-size: 1.5em;
19  18 }
19  +
20  20 .App-intro {
21  - font-size: large;
21  +
22  + font-size: 2em;
22  23 }
23  +
23  25 @keyframes App-logo-spin {
24  26 from { transform: rotate(0deg); }
25  27 to { transform: rotate(360deg); }
26  28 }
27  29 .card {
28  30 display: inline-block;
29  31 text-align: center;
30  32 width: 3em;
31  - font-size: 2em;
33  +
34  + font-size: 3em;
32  35 box-shadow: 0 4px 8px 0 rgba(0,0,0,0.2);
33  36 margin: 1em;
34  37 user-select: none;
35  38 }
36  39 .activeCard {
37  40 color: blue;
38  - background: pink;
39  - }
41  + background: rgb(153, 208, 240);
42  + }
43  +

```

## 7. ทดลองปรับเปลี่ยนสีพื้นหลังตัวอักษร

▼ 13 src/App.css		
...	...	@@ -1,6 +1,6 @@
1	1	.App {
2	2	text-align: center;
3	-	background-color: chartreuse;
3	+	background-color: rgb(0, 153, 255);
4	4	height: 100vh;
5	5	}
6	6	.App-logo {
7	7	animation: App-logo-spin infinite 20s linear;
8	8	height: 80px;
9	9	}
10	10	.App-header {
11	11	background-color: #222;
12	12	height: 150px;
13	13	padding: 20px;
14	14	color: white;
15	15	}
16	16	.App-title {
17	17	font-size: 1.5em;
18	18	}
19	-	
20	19	.App-intro {
21	-	
22	20	font-size: 2em;
23	21	}
24	-	
25	22	@keyframes App-logo-spin {
26	23	from { transform: rotate(0deg); }
27	24	to { transform: rotate(360deg); }
28	25	}
29	26	.card {
30	27	display: inline-block;
31	-	text-align: center;
28	+	
29	+	text-align: unset;
32	30	width: 3em;
33	-	
34	31	font-size: 3em;
35	32	box-shadow: 0 4px 8px 0 rgba(0,0,0,0.2);
36	33	margin: 1em;
37	34	user-select: none;
38	35	}
39	36	.activeCard {
40	37	color: blue;
41	38	background: rgb(153, 208, 240);
42	-	}
43	-	
39	+	background-color: darkorange;
40	+	}

## 8. ทดลองสร้างเกมส์ Logic ที่สามารถนับรอบและแสดงผลพัธได้

```

src/CharacterCard.js
... @@ -1,26 +1,31 @@
1 - import React, { useState, useEffect, useRef } from 'react';
2 -
3 - export default function CharacterCard(props) {
4 -   const [active, setActive] = useState(false);
5 -   const attemptRef = useRef(props.attempt);
6 -
7 -   const activate = () => {
8 -     if(!active){
9 -       setActive(true)
10 -      props.activationHandler(props.value)
11
12 + import React, { Component } from 'react';
13 + export default class CharacterCard extends Component {
14 +   constructor(props){
15 +     super(props)
16 +     this.state = {
17 +       active: false,
18
19   7   }
20   8   }
21
22 -   useEffect(() => {
23 -     if(attemptRef.current !== props.attempt){
24 -       setActive(false);
25 -       attemptRef.current = props.attempt
26
27 +   activate = () => {
28 +     if(!this.state.active){
29 +       this.props.activationHandler(this.props.value)
30 +       this.setState({active: true})
31 +     }
32 +   }
33
34 +   componentDidUpdate(prevProps){
35 +     if(prevProps.attempt !== this.props.attempt){
36 +       this.setState({active: false})
37 +     }
38
39   21   }
40   22   })
41
42 -   const className = `card ${active ? 'activeCard' : ''}`
43
44 +   render(){
45 +     let className = `card ${this.state.active ? 'activeCard': ''}`
46 +     return (
47
48 -     <div className={className} onClick={activate}>{props.value}</div>
49 -   )
50
51 +     <div className={className} onClick={this.activate}>
52 +       {this.props.value}</div>
53 +   )
54 + }
55
56   31 }

```

```

✓ 90 src/WordCard.js
...  ... @@ -1,48 +1,54 @@
1 - import React, { useState } from 'react';
1 + import React, { Component } from 'react';
2 + import CharacterCard from "../CharacterCard";
2 3 import _ from 'lodash';
3 -
4 - import CharacterCard from './CharacterCard';
5 -
6 - const prepareStateFromWord = given_word => {
4 + const prepareStateFromWord = (given_word) => {
7 5 let word = given_word.toUpperCase()
8 6 let chars = _.shuffle(Array.from(word))
9 7 return {
10 - word,
11 - chars,
12 - attempt: 1,
13 - guess: '',
14 - completed: false
8 + word,
9 + chars,
10 + attempt: 1,
11 + guess: [],
12 + completed: false
15 13 }
16 - }
17 -
18 - export default function WordCard(props){
19 -
20 - const [state, setState] = useState(prepareStateFromWord(props.value))
21 -
22 - const activationHandler = c =>{
23 - console.log(`${c} has been activated`)
24 -
25 - let guess = state.guess + c
26 - setState({...state, guess})
27 -
28 - if(guess.length == state.word.length){
29 - if(guess==state.word){
30 - console.log('yeah!')
31 - setState({...state, completed: true})
32 - }else{
33 - console.log('reset, next attempt')
34 - setState({...state, guess: '', attempt: state.attempt + 1})
35 - }
36 - }

```

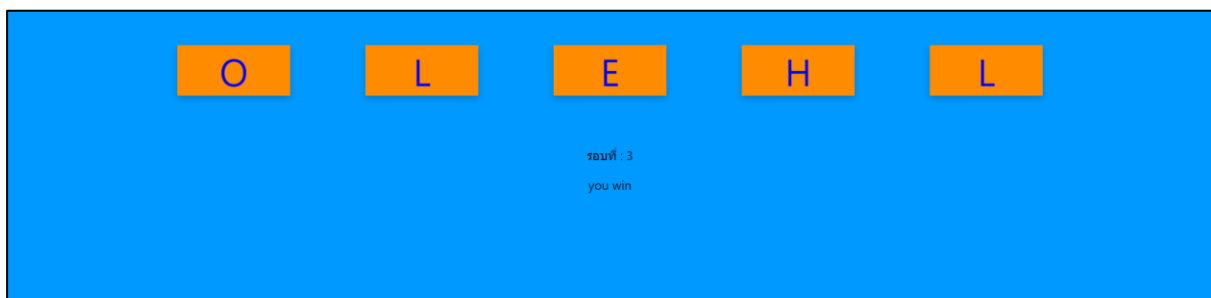
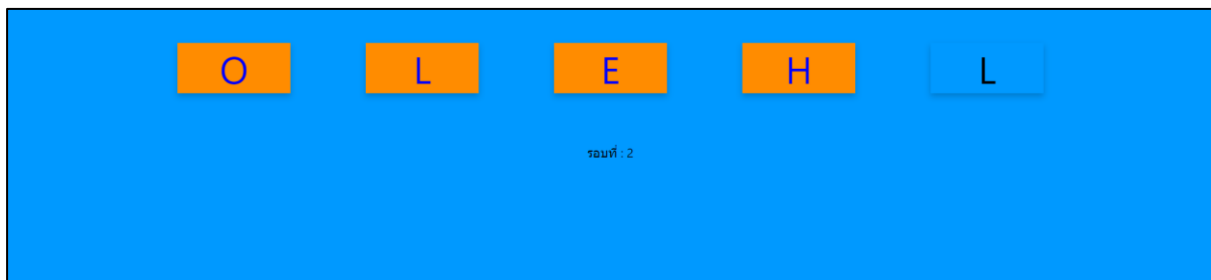
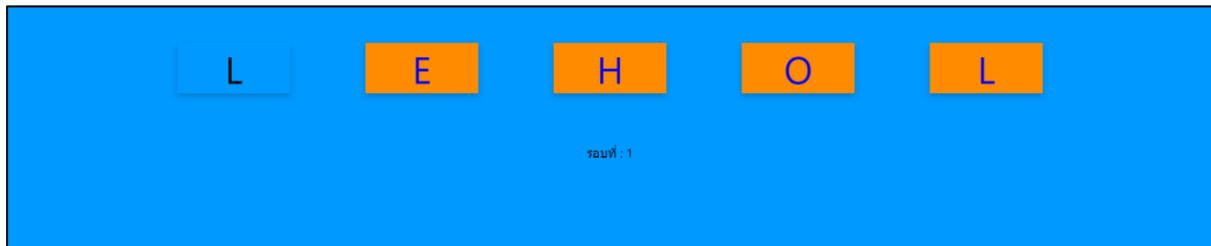
```

14 +   }
15 +
16 +   export default class WordCard extends Component {
17 +     constructor(props){
18 +       super(props)
19 +       this.state = prepareStateFromWord(this.props.value)
37 20   }
38 -
39 -   return(
40 -     <div>
41 -       {
42 -         state.chars.map((c, i) =>
43 -           <CharacterCard value={c} key={i} activationHandler={activationHandler} attempt={state.attempt}/>
44 -         )
45 -       }
46 -     </div>
47 -   )
48 - } ⊖
21 +   activationHandler = (c) => {
22 +
23 +     let guess = [...this.state.guess, c]
24 +     this.setState({guess})
25 +
26 +     if(guess.length == this.state.chars.length){
27 +       if(guess.join('').toString() == this.state.word){
28 +         this.setState({guess: [], completed: true})
29 +       }else{
30 +         this.setState({guess: [], attempt: this.state.attempt + 1})
31 +       }
32 +     }
33 +   }
34 + }
35 +
36 + render(){
37 +   return(
38 +     <div className="App">
39 +       {
40 +         Array.from(this.state.chars).map((c, i) => <CharacterCard value = {c} key = {i}
41 +
42 +           attempt={this.state.attempt}
43 +           activationHandler = {this.activationHandler} />
44 +         )
45 +       }
46 +
47 +       <p>รอบที่ : {this.state.attempt}</p>
48 +       <p>{this.state.completed? "you win" : ""}</p>
49 +     </div>
50 +   );
51 + }
52 +
53 +
54 + } ⊖

```



แสดงตัวอย่างผลลัพธ์เกม Logic ที่ออกแบบ



ลักษณะของเกมก็คือ ถ้ากดทายคำศัพท์ผิดเกมส้จะเพิ่มจำนวนรอบขึ้นเรื่อย ๆ และเกมส้จะหยุดเมื่อ  
กดทายคำศัพท์ถูกต้องแล้ว พร้อมแสดงผลลัพธ์เตือนว่า Win