



hello_vue

This is a beginning for vue.js

Summary: This document is the subject for vue.js @42Tokyo.

Contents

I	Instructions	2
II	Foreword	3
III	Exercise 00 : Hello 42!	4
IV	Exercise 01 : Hello binding	5
V	Exercise 02 : Button	6
VI	Exercise 03 : User input	8
VII	Exercise 04 : Components	9
VIII	Exercise 05 : Props	10

Chapter I

Instructions

- Only this page will serve as reference; do not trust rumors.
- Watch out! This document could potentially change up to an hour before submission.
- Make sure you have the appropriate permissions on your files and directories.
- You have to follow the submission procedures for every exercise.
- Your exercises will be checked and graded by your fellow classmates.
- You cannot leave any additional file in your directory than those specified in the subject.
- Got a question? Ask your peer on the right. Otherwise, try your peer on the left.
- Your reference guide is called `Google / man / the Internet /`
- Examine the examples thoroughly. They could very well call for details that are not explicitly mentioned in the subject...
- your vue version should be $\geq 2.5.16$. (we will use <https://unpkg.com/vue> which is v2.6.12)


Chapter II

Foreword

Vue is a progressive framework for building user interfaces. Unlike other monolithic frameworks, Vue is designed from the ground up to be incrementally adoptable. The core library is focused on the view layer only, and is easy to pick up and integrate with other libraries or existing projects. On the other hand, Vue is also perfectly capable of powering sophisticated Single-Page Applications when used in combination with modern tooling and supporting libraries.

Chapter III


Exercise 00 : Hello 42!

	Exercise 00
Only the best know how to display Hello 42!	
Turn-in directory : <i>ex00/</i>	
Files to turn in : index.html	
Allowed functions : None	

- Access to this site below
- <https://bit.ly/2KPRULC>
- If the url above doesn't work copy paste the link below into a search engine.
- <https://codesandbox.io/s/github/vuejs/vuejs.org/tree/master/src/v2/examples/vue-20-hello-world>
- Replace 'Hello Vue!' by 'Hello 42!'.
- replace the the value inside title tag to "My first Vue app for 42".
- Turn-in the modified index.html.

Chapter IV

Exercise 01 : Hello binding

	Exercise 01
How to interact with the UI?	
Turn-in directory : <i>ex01/</i>	
Files to turn in : index.html	
Allowed functions : None	

- Access to this site below
- <https://bit.ly/2KPRULC>
- If the url above doesn't work copy paste the link below into a search engine.
- <https://codesandbox.io/s/github/vuejs/vuejs.org/tree/master/src/v2/examples/vue-20-hello-world>
- Create a 'seen' property inside the data object.

```
<script>
var app = new Vue({
  el: '#app',
  data: {
    message: 'Hello Vue!',
    seen: true
  }
})
</script>
```


- If seen property is set to true, it should display "Hello Vue!".
- If seen property is set to false, it shouldn't display anything.
- Turn-in the modified index.html.



v-if

Chapter V

Exercise 02 : Button

	Exercise 02
Interact by clicking button	
Turn-in directory : <i>ex02/</i>	
Files to turn in : index.html	
Allowed functions : None	

- Access to this site below
- <https://bit.ly/2KPRULC>
- If the url above doesn't work copy paste the link below into a search engine.
- <https://codesandbox.io/s/github/vuejs/vuejs.org/tree/master/src/v2/examples/vue-20-hello-world>
- Create a 'seen' property inside the data object.

```
<script>
var app = new Vue({
  el: '#app',
  data: {
    message: 'Hello Vue!',
    seen: true
  }
})
</script>
```

- Create a button tag with type="button" inside the index.html and the button should contain this message "SHOW".
- Create a button tag with type="button" inside the index.html and the button should contain this message "HIDE".
- If the button "SHOW" is clicked , it should display "Hello Vue!".
- If the button "HIDE" is clicked , it should hide the message "Hello Vue!".


- Turn-in the modified index.html.



`v-on:click`

Chapter VI

Exercise 03 : User input

	Exercise 03
Interact with user input	
Turn-in directory : <i>ex03/</i>	
Files to turn in : index.html	
Allowed functions : None	


- Access to this site below
- <https://bit.ly/2KPRULC>
- If the url above doesn't work copy paste the link below into a search engine.
- <https://codesandbox.io/s/github/vuejs/vuejs.org/tree/master/src/v2/examples/vue-20-hello-world>
- Create a input form.
- if user type in the input form, the message displayed on the screen should change.
- Turn-in the modified index.html.



v-model

Chapter VII

Exercise 04 : Components

	Exercise 04
What are components	
Turn-in directory : <i>ex04/</i>	
Files to turn in : index.html	
Allowed functions : None	


- Access to this site below
- <https://bit.ly/2KPRULC>
- If the url above doesn't work copy paste the link below into a search engine.
- <https://codesandbox.io/s/github/vuejs/vuejs.org/tree/master/src/v2/examples/vue-20-hello-world>
- Create a component called "hello". this component should contain a template with this value inside : "`<h1>Hello Vue!</h1>`".
- Add this component inside the `<div id="app">` section.
- Turn-in the modified index.html.



<https://jp.vuejs.org/v2/guide/components.html>

Chapter VIII

Exercise 05 : Props

	Exercise 05
Props Props Props	
Turn-in directory : <i>ex05/</i>	
Files to turn in : index.html	
Allowed functions : None	

- Access to this site below
- <https://bit.ly/2KPRULC>
- If the url above doesn't work copy paste the link below into a search engine.
- <https://codesandbox.io/s/github/vuejs/vuejs.org/tree/master/src/v2/examples/vue-20-hello-world>
- replace the content of the script tag by this value below

```
Vue.component('series-item', {
  props: ['series'],
  template: '<h2>{{ series.text }}</h2>'
})

var app = new Vue({
  el: '#app',
  data: {
    NetflixList: [
      { id: 0, text: "Queen's Gambit" },
      { id: 1, text: "Altered Carbon" },
      { id: 2, text: "Black Mirror" }
    ]
  }
})
```

- Modify the content inside `<div id="app">` so that it print's the value inside `NetflixList` property by using `'series-item'` component.
- Turn-in the modified `index.html`.



<https://jp.vuejs.org/v2/guide/components.html>