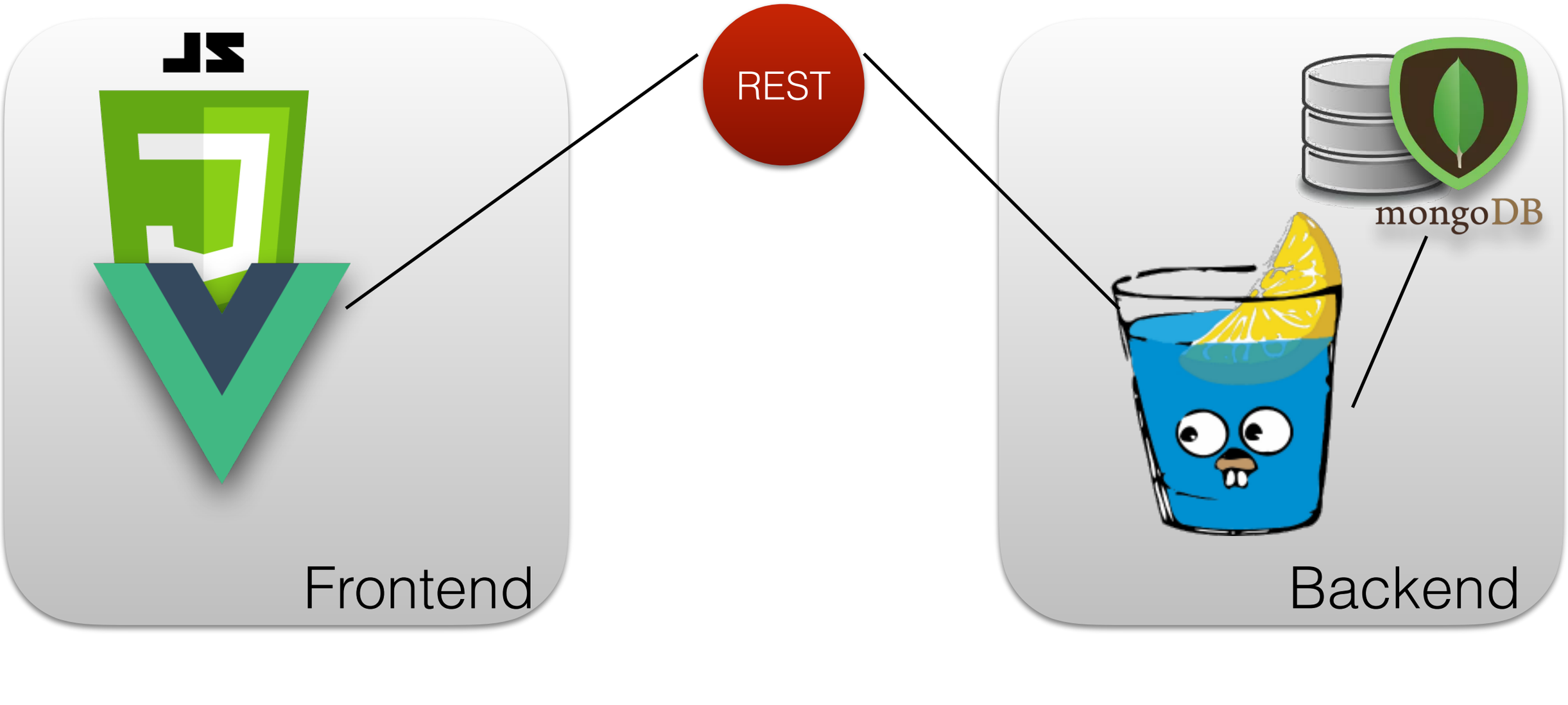
Gin Gonic/ Vue.js

Presentation of the two stunning Webframeworks in GoLang & JavaScript

by Alex Schübl, David Bochan, Nadin-Katrin Apel
in SS 2018

Table of Contents

Separation of Backend/ Frontend



Recap from last week

https://www.youtube.com/watch?v=5NfuIIWhQxI

HelloWorld ++ = Calculator App

Table of Contents

Table of Contents

Get Started


```
<script src="https://cdn.jsdelivr.net/npm/vue/dist/vue.js"></script>
```

Create a new Vue Instance

```
var app = new Vue({
    el: '#app',
    data: {
        message: 'Hello Vue!'
    }
})
```

Get Started

youtube.com/watch?v=cmNMcTQv4Dw

```
<script src="https://cdn.jsdelivr.net/npm/vue/dist/vue.js"></script>
```

Exercise 1

- Create a new Vue Instance
- Install Vue developer tools as addOn in your browser
- create some data state attributes in the Vue instance
- play around in the browser console with your app data state in the dev tools & console

```
var app = new Vue({
    el: '#calc-app',
    data: {
        name: 'Hero'
    }
})
```

Some Vocabulary

- Declarative Rendering
- Reactivity
- Interpolations
  - Text
  - Raw HTML
  - HTML attributes (eg. value, src)
- Directives

Table of Contents

Interpolation Basics

JavaScript Expressions

```
{{name}}
{{number + 1}}
{{date ? 'yes' : 'no'}}
<input type="text" v-on:input="input => name = input.target.value">
```

Directives

```
<input type="text" v-on:input="input => name = input.target.value">
<img v-bind:src="image">
<div v-html="rawHTML">
<input type="text" v-model="userInput">
```

class binding

```
<div class="static" :class="{ active: isActive }"></div>
<div :class="[activeClasses, errorClasses]"></div>
```

style binding

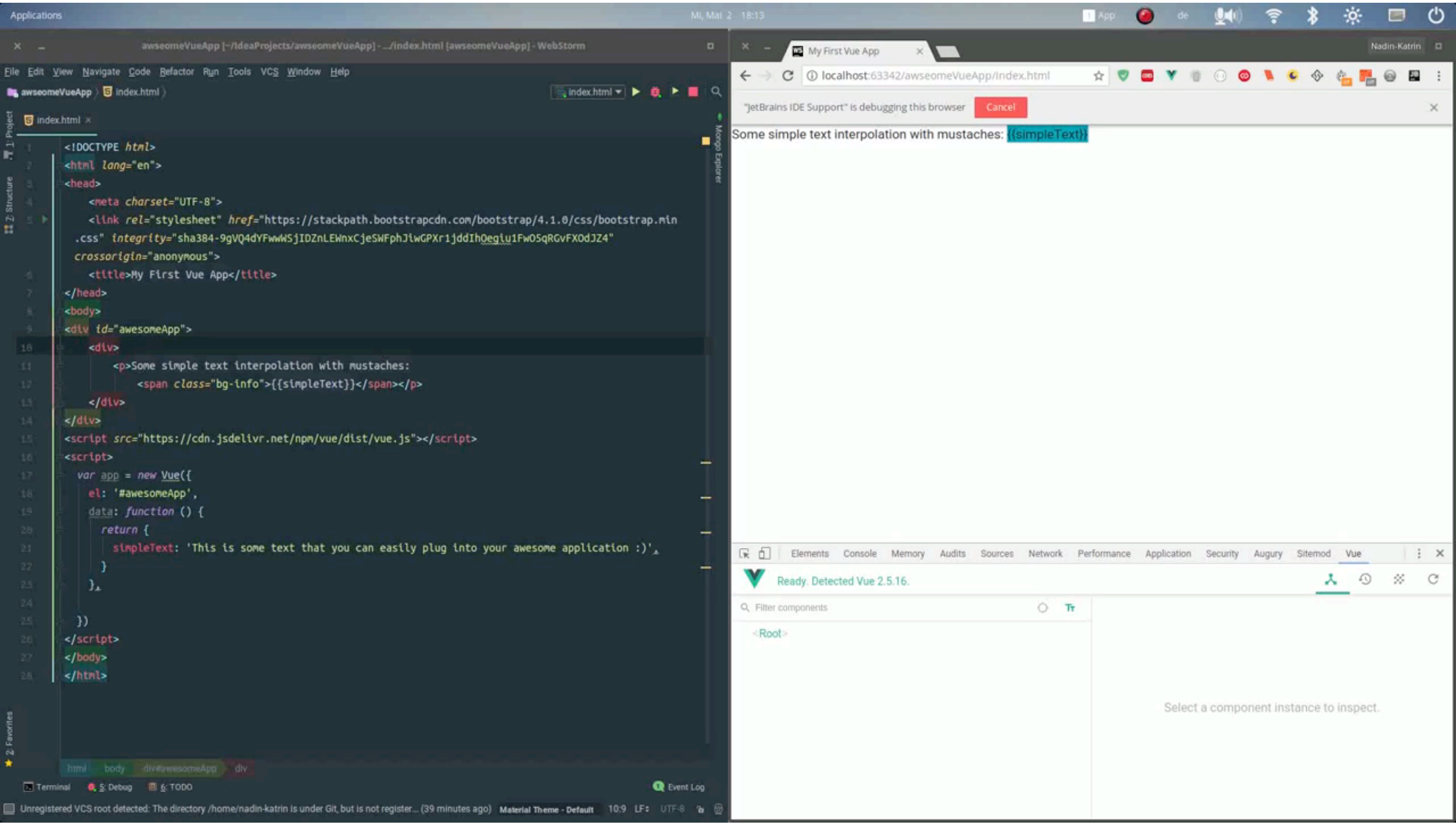
```
<div :style="{ width: points * 10 + '%', background: 'green' }"></div>
```

Shortages

v-on:input	*@input*
v-on:change	*@change*
v-on:click	*@click*
v-bind:value	*:value*
v-bind:src	*:src*

2.1

Interpolation Basics



youtube.com/watch?v=qQtWZNWSU-w

Interpolation Basics

JavaScript Expressions

```
{{name}}
{{number + 1}}
{{date ? 'yes' : 'no'}}
<input type="text" v-on:input="input => name = input.target.value">
```

Directives

```
<input type="text" v-on:input="input => name = input.target.value">
<img v-bind:src="image">
<div v-html="rawHTML">
<input type="text" v-model="userInput">
```

class binding

```
<div class="static" :class="{ active: isActive }"></div>
<div :class="[activeClasses, errorClasses]"></div>
```

style binding

```
<div :style="{ width: points * 10 + '%', background: 'green' }"></div>
```

Shortages

v-on:input	*@input*
v-on:change	*@change*
v-on:click	*@click*
v-bind:value	*:value*
v-bind:src	*:src*

2.2

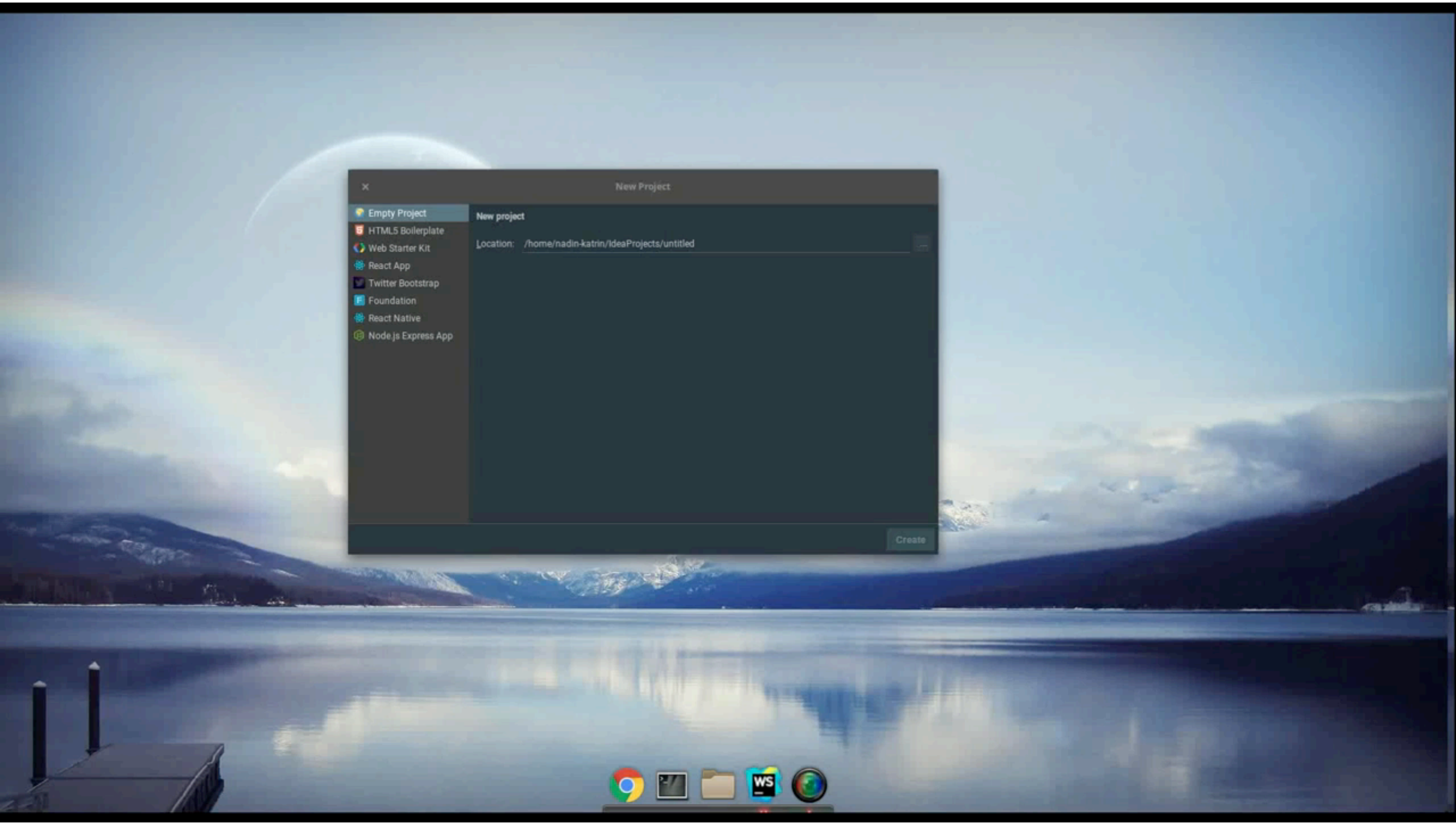
Exercise 2 a)

Become a Calculus Master

Calculation training completed at 5/3/2018 19:9:30 by Mathe Meister Mathe Meister

```
var getTimeStamp = function () {
    var today = new Date()
    return today.toLocaleDateString() + ' ' + today.getHours() + ':' + today.getMinutes() + ':' + today.getSeconds()
}
var randomNumberGenerator = function () {
    return parseInt(Math.random() * (99) + 1)
}
```

Exercise 2: Solution



https://www.youtube.com/watch?v=R9a9C2IeHiQ

Methods/ Events

```
<template>
    <input type="Number" v-model="a">
    <input type="Number" v-model="b">
    <button @click="result = a+b"></button>
    <input type="Number" v-bind:value="result">
</template>

<script>
    export default {
        name: "index",
        data() {
            return {
                a:23,
                b:234,
                result: null
            }
        },
    }
</script>
```

event: v-on:click = @click

1) inline method

2.2

Methods/ Events

event: v-on:click = @click

2) external method

```
<div id=“app”>
    <input type="Number" v-model="a">
    <input type="Number" v-model="b">
    <button @click="calculate()"></button>
    <input type="Number" v-bind:value="result">
</div>

<script>
    var app = new Vue({
        el: “#app",
        data() {
            return {
                a:23,
                b:234,
                result: null
            }
        },
        methods: {
            calculate(){
                this.result =  this.a + this.b
            }
        }
    })
</script>
```

2.2

Methods

event: v-on:click = @click

2) external method
with parameters

```
<div id=“app”>
    <input type="Number" v-model="a">
    <input type="Number" v-model="b">
    <button @click="result = calculate(a,b)"></button>
    <input type="Number" v-bind:value="result">
</div>
<script>
    var app = new Vue({
        el: “#app",
        data() {
            return {
                a:23,
                b:234,
                result: null
            }
        },
        methods: {
            calculate(a,b){
                return a + b
            }
        }
    }
</script>
```

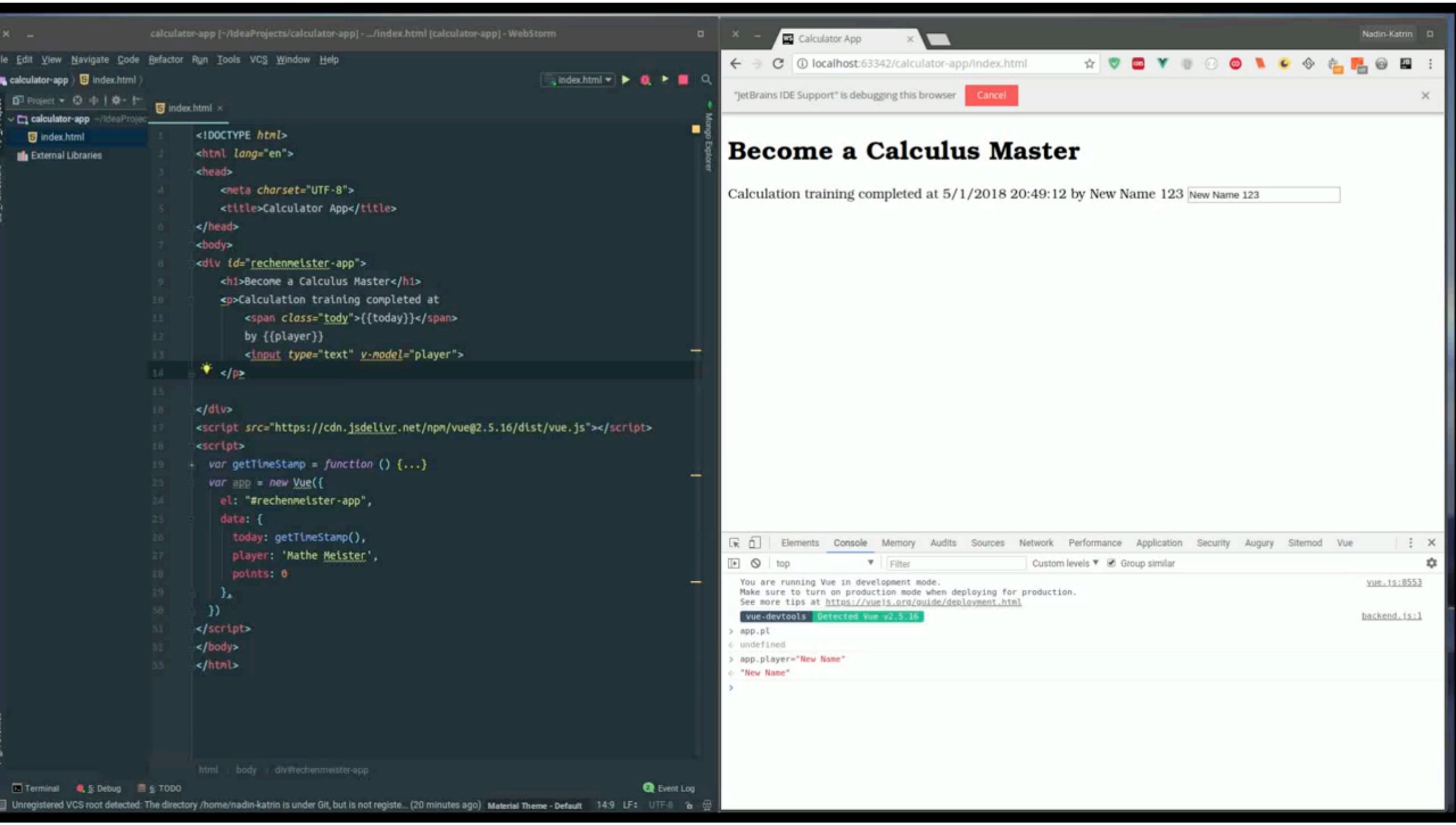
2.2

Exercise 2 b)

https://youtu.be/9L3PLbnJ120

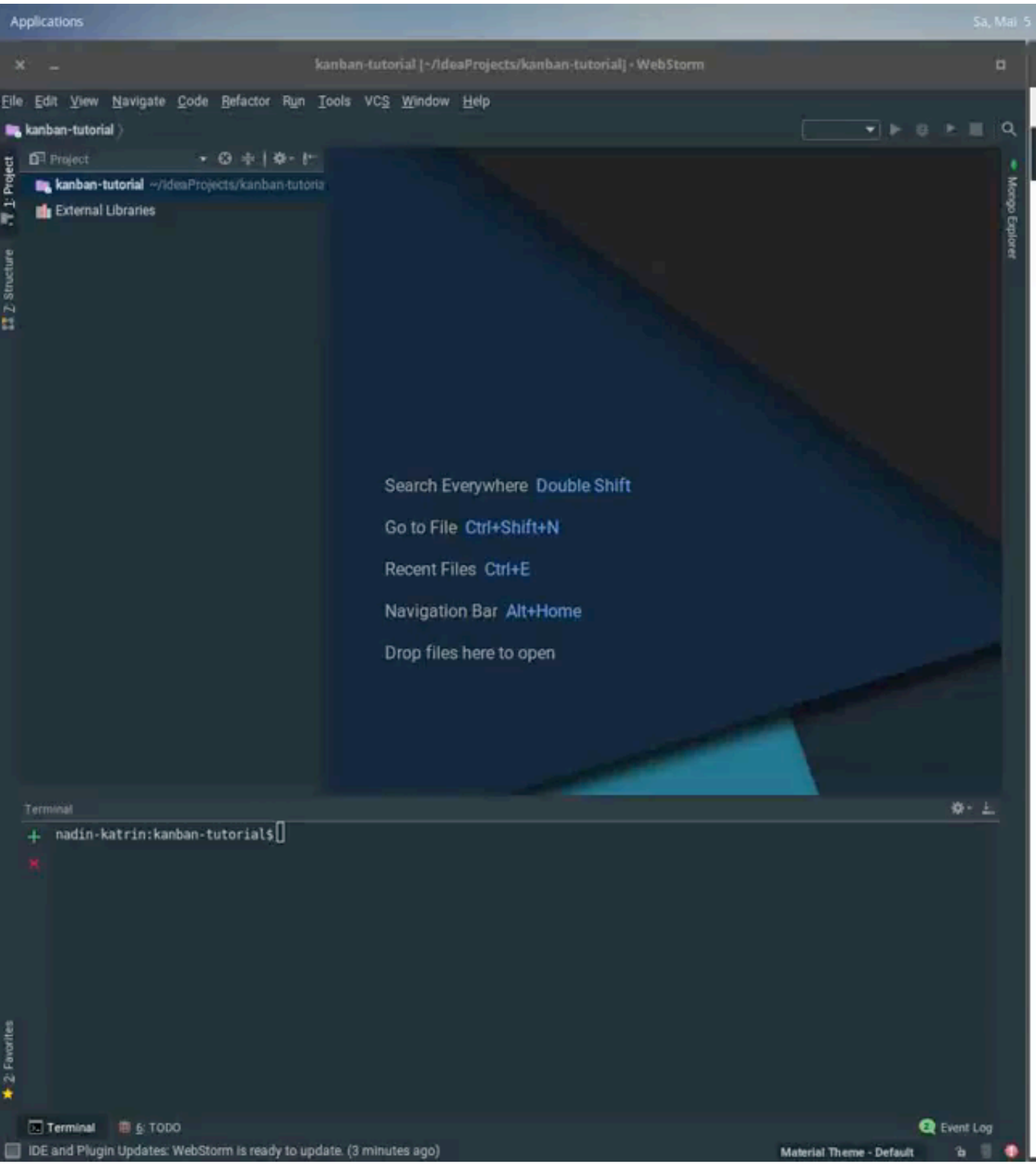
```
var getTimeStamp = function () {
    var today = new Date()
    return today.toLocaleDateString() + ' ' + today.getHours() + ':' + today.getMinutes() + ':' + today.getSeconds()
}
var randomNumberGenerator = function () {
    return parseInt(Math.random() * (99) + 1)
}
```

Exercise 2: Solution



https://www.youtube.com/watch?v=u0Memcp9gAw

Recap: data binding kanban board



youtu.be/WitYrzpW-Pw

2.3

Conditional Rendering (v-if/ v-else, v-show)

```
<p v-if="date">Sven goes on a date with Juliet</p>
<p v-else-if="!date && practice">Sven goes to soccer practice</p>
<p v-else="!date && !practice">Sven watches Netflix lonely at home</p>
<p v-show="date">Sven is really happy :) </p>
```

https://www.youtube.com/watch?v=R9a9C2IeHiQ

2.3

Conditional Rendering (v-if/ v-else, v-show)

parent.vue

```vue
<template>
    <v-card class="pa-3">
        <child-component name="Jessi" :age="4" place="New York"></child-component>
        <child-component name="Jonas" :age="24" place="Stuttgart"></child-component>
    </v-card>
</template>
```

Jessi is 4 years old and lives in New York

Our child has a very cute name! 💖

Jonas is 24 years old and lives in Stuttgart

child.vue

```vue
<template>
    <div class="elevation-1 pa-1 ma-1">
        <p>{{name}} is {{age}} years old and lives in {{place}}</p>
        <p v-if="name === 'Jessi'">Our child has a very cute name! 💖</p>
    </div>
</template>
```

```html
▼<div class="elevation-1 pa-1 ma-1"> == $0
    <p>Jessi is 4 years old and lives in New York</p>
    <p>Our child has a very cute name! 💖</p>
  </div>
▼<div class="elevation-1 pa-1 ma-1">
    <p>Jonas is 24 years old and lives in Stuttgart</p>
    <!---->
  </div>
</div>
```

v-if

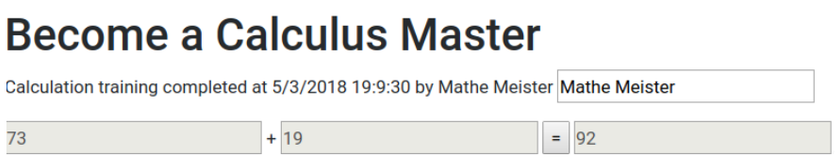
child.vue

```vue
<template>
    <div class="elevation-1 pa-1 ma-1">
        <p>{{name}} is {{age}} years old and lives in {{place}}</p>
        <p v-show="name === 'Jessi'">Our child has a very cute name! 💖</p>
    </div>
</template>
```

```html
▼<div class="elevation-1 pa-1 ma-1"> == $0
    <p>Jessi is 4 years old and lives in New York</p>
    <p>Our child has a very cute name! 💖</p>
  </div>
▼<div class="elevation-1 pa-1 ma-1">
    <p>Jonas is 24 years old and lives in Stuttgart</p>
    <p style="display: none;">Our child has a very cute name! 💖</p>
  </div>
```

v-show

Exercise hints




```
head
  link  https://stackp...otstrap/4.1.0/css/bootstrap.min.css
  meta
  title  Calculator App
body
  div#rechenmeister-app
    div.header
      h1  Become a Calculus Master
    p  Calculation training completed at by {{player}}
      span  {{today}}
      input
  div.calculator  {{op}}
    input
    input
    button  =
    input
    button  Test
    input
    p  {{result === guessedResult ? 'yay - success' : 'oh...
    button  New Game
script  https://cdn.js...livr.net/npm/vue@2.5.16/dist/vue.js
```

```
script
  λ getTimeStamp()
  λ randowmNumberGenerator()
  v app
    p "#rechenmeister-app"
    p el
    p data
      today
      player
      points
      a
      b
      result
      guessedResult
      op
    p methods
      m recalculate(): void
```

Become a Calculus Master

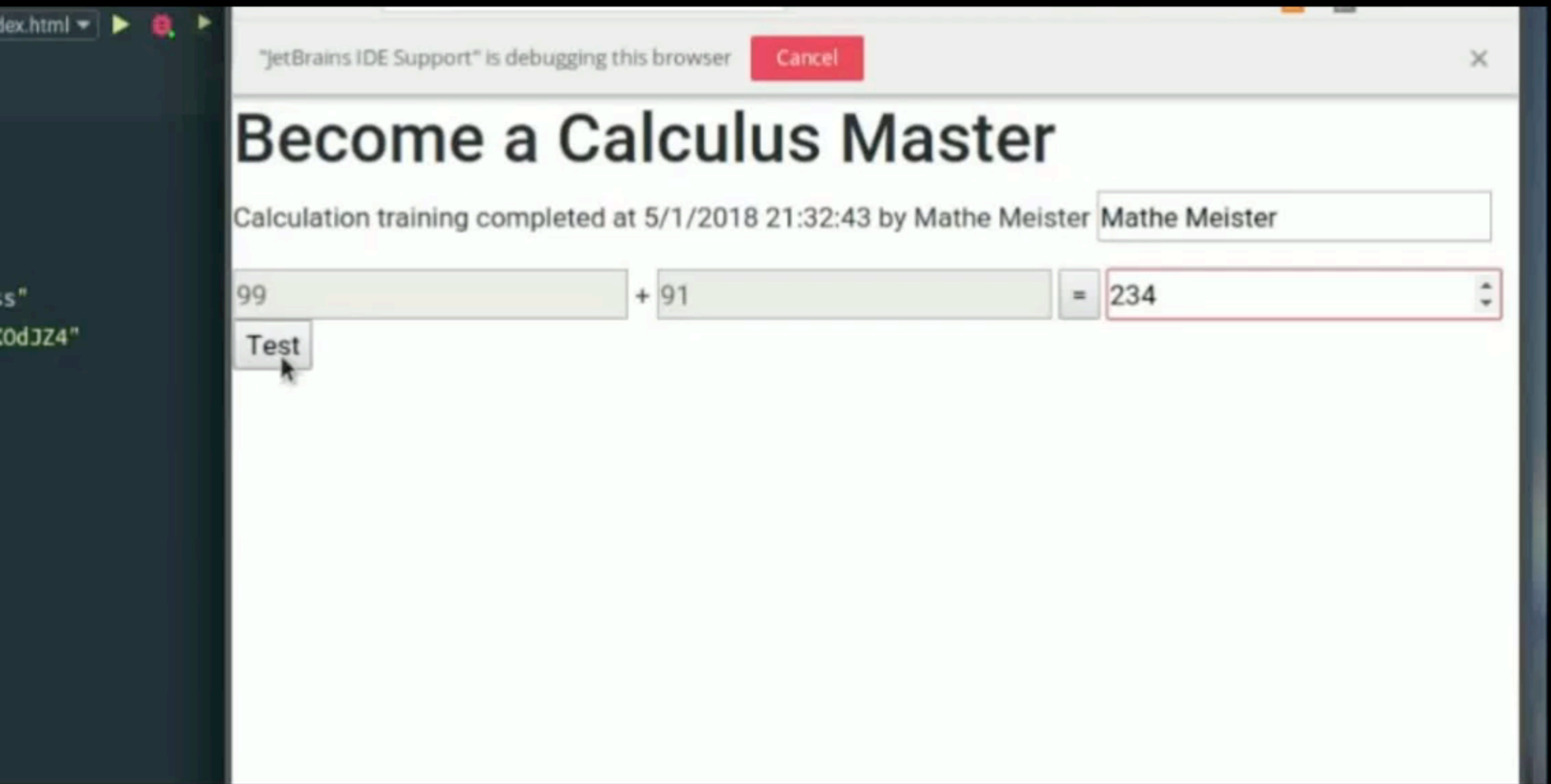
Calculation training completed at 5/3/2018 19:9:30 by Mathe Meister Mathe Meister

73 + 19 = 92 Test 92

yay - success

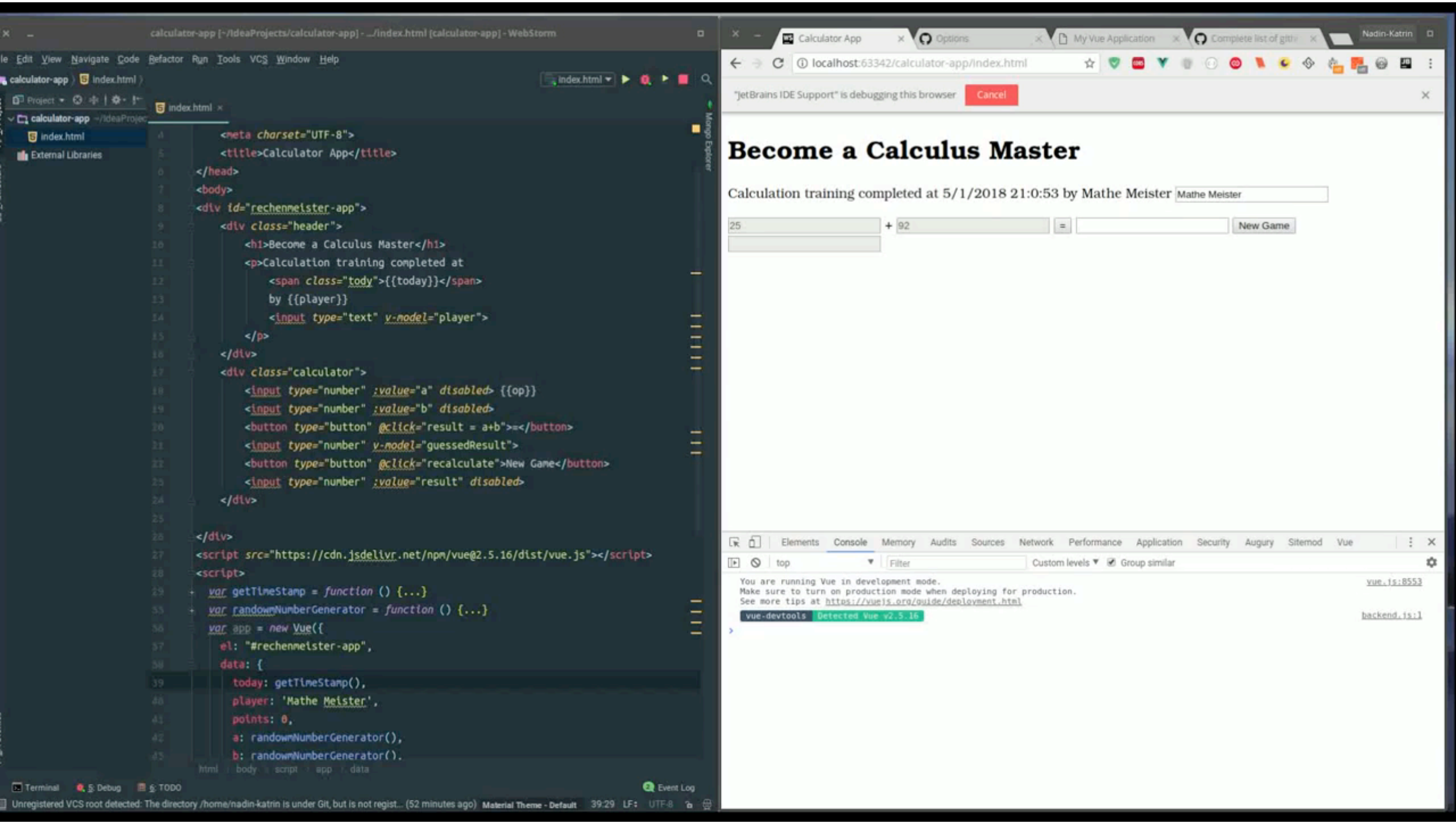
New Game

Exercise 2 c)



https://youtu.be/QyRj7GBLmLc

Exercise 2 c) Solution



https://youtu.be/KryfMd6vrE4

2.4 Directives: loops (v-for)

```
<template>
    <li v-for="(player,index) in resultArray">{{player}}</li>
    <li v-for="(player,key, index) in resultObject" :key="player.key">{{player[key]}}</li>
</template>
```

```
data() {
    return {
        resultArray: [
            {name: 'hero', points: 97},
            {name: 'john', points: 123},
            {name: 'jess', points: 45},
        ],
        resultObject: {
            hero: {points: 97},
            john: {points: 123},
            jess: {points: 45},
        }
    }
},
```

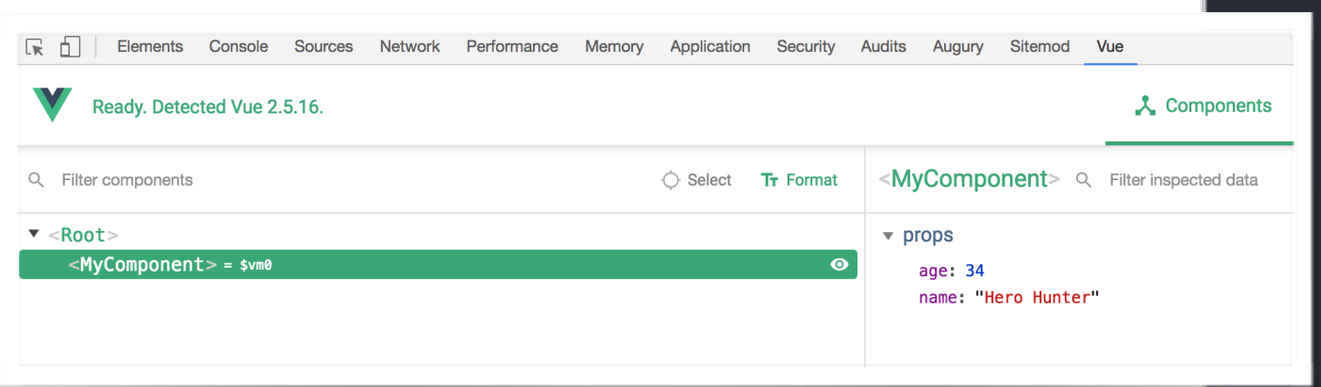
https://youtu.be/u0Memcp9gAw

Table of Contents

3. Components

```
<div id="app">
    <my-component :age="34" name="Hero Hunter"></my-component>
</div>
```

```
Vue.component('my-component', {
        props: ['age', 'name'],
        template: `<p>{{name}} is {{age}} years old</p>`,
    }
)
```



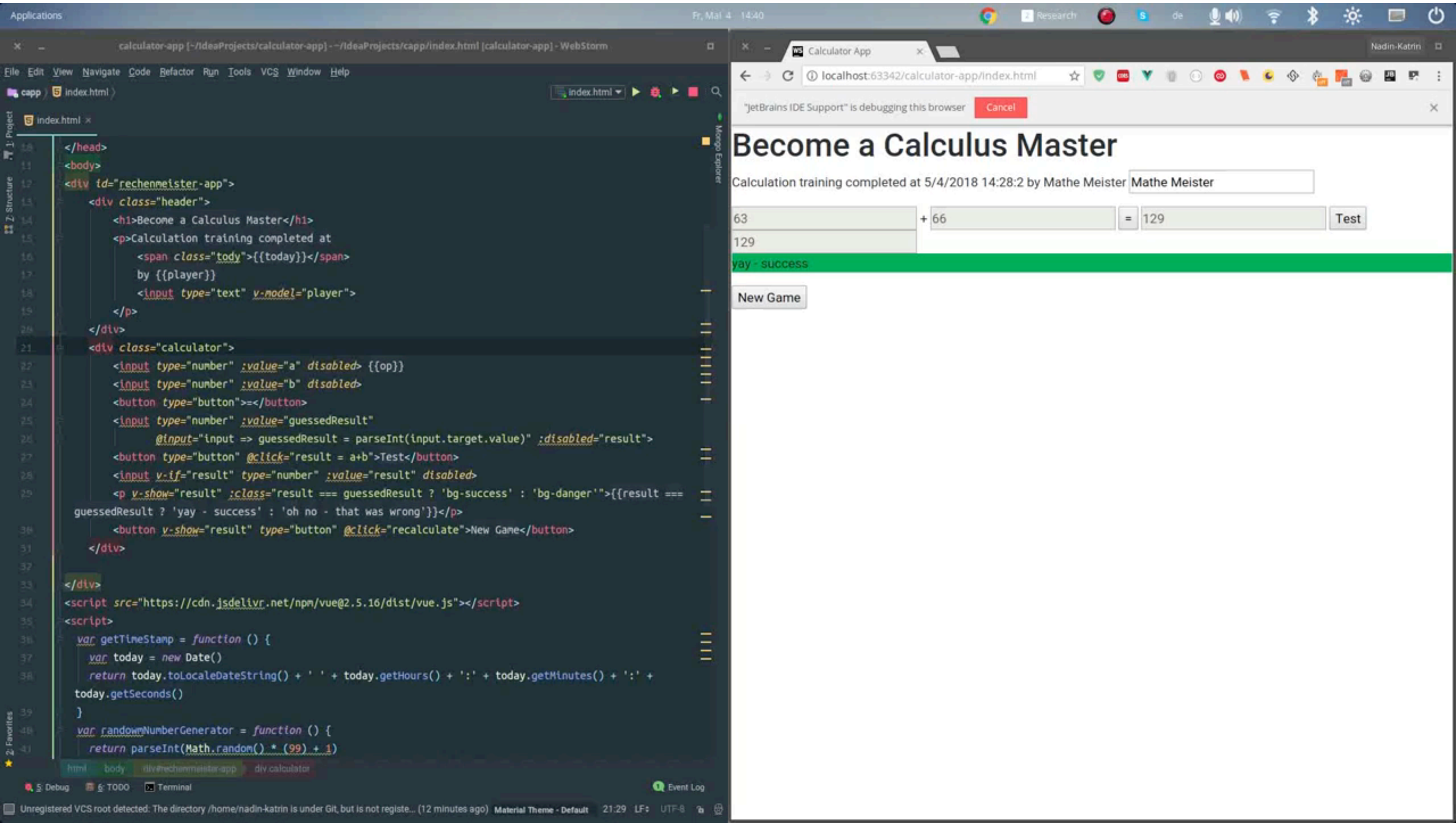

```
Vue.component('my-component', {
        props: {
            age:
                {type: Number, required: false, default: 23},
            name:
                {type: String, required: true},
    },
    template: `<p>{{name}} is {{age}} years old</p>`,
```

3. Exercise Components

props: [‘player’, ‘points’, ‘maxRounds’, ‘date’]

https://youtu.be/fTtPfba5BIg

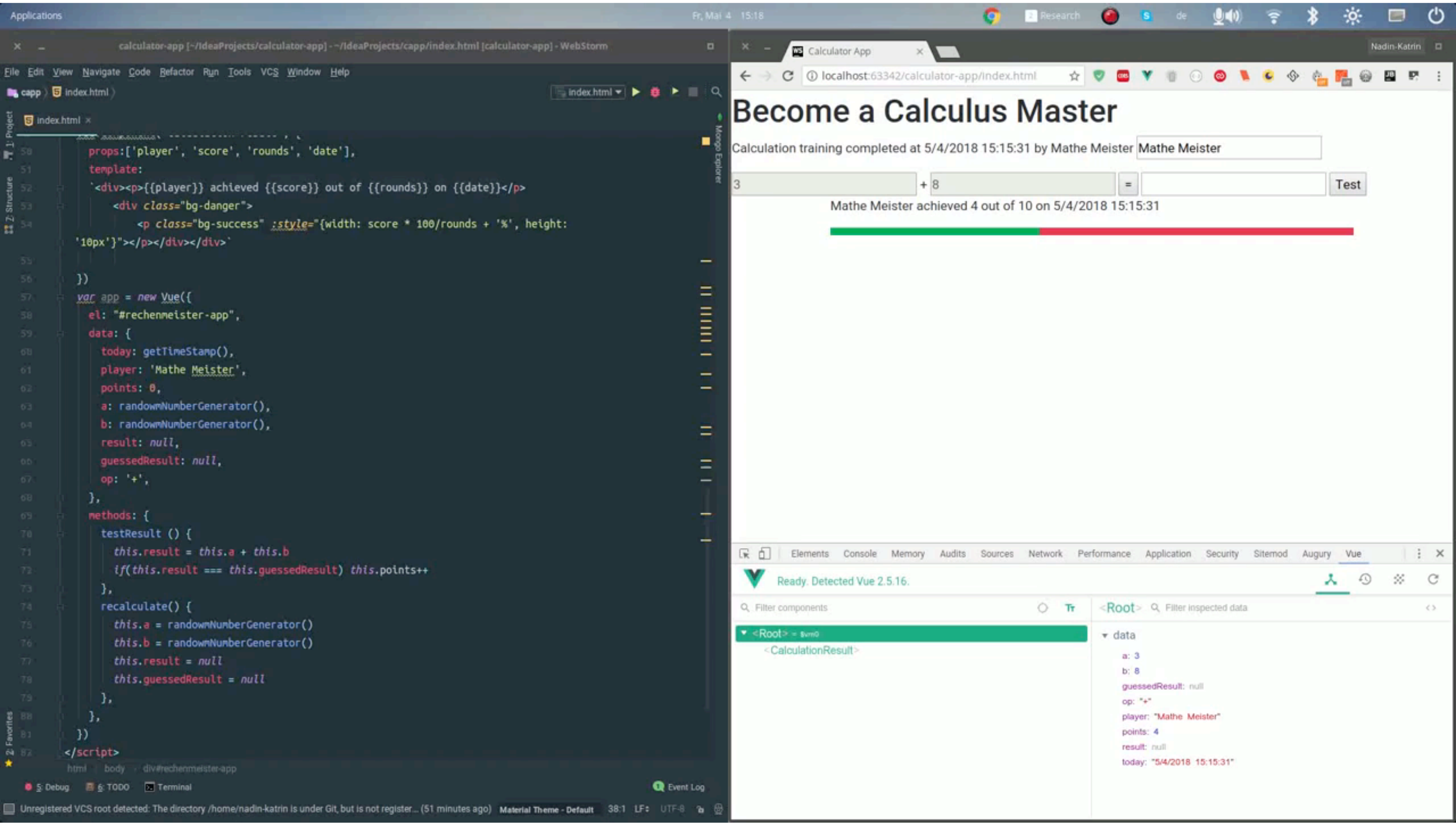
3. Exercise Components: Solution



3. Exercise Components, loops, conditions

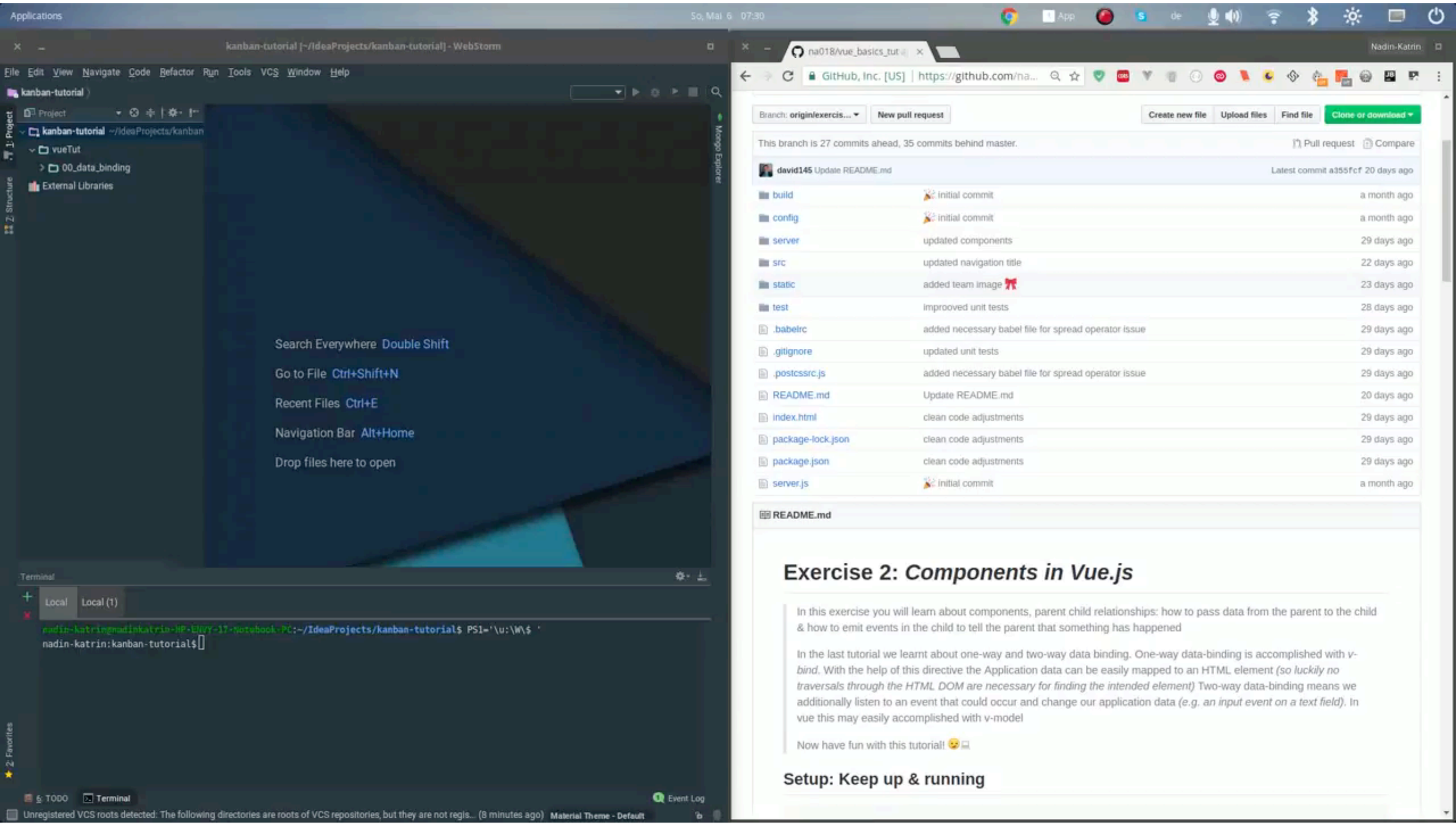
https://youtu.be/8RCfCICfmTU

3. Exercise Components, loops, conditions: solution



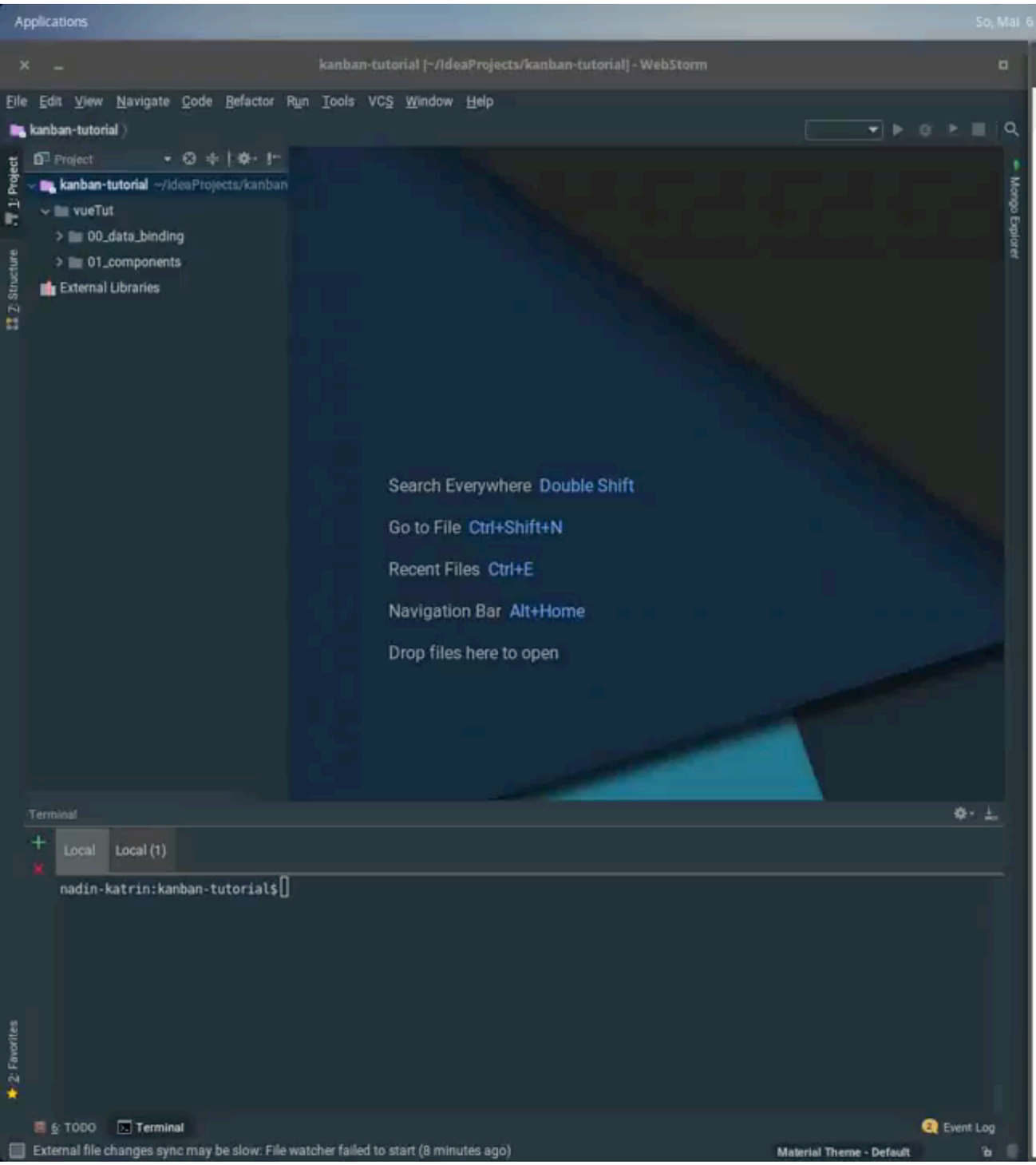
https://youtu.be/GKsGREqHjPw

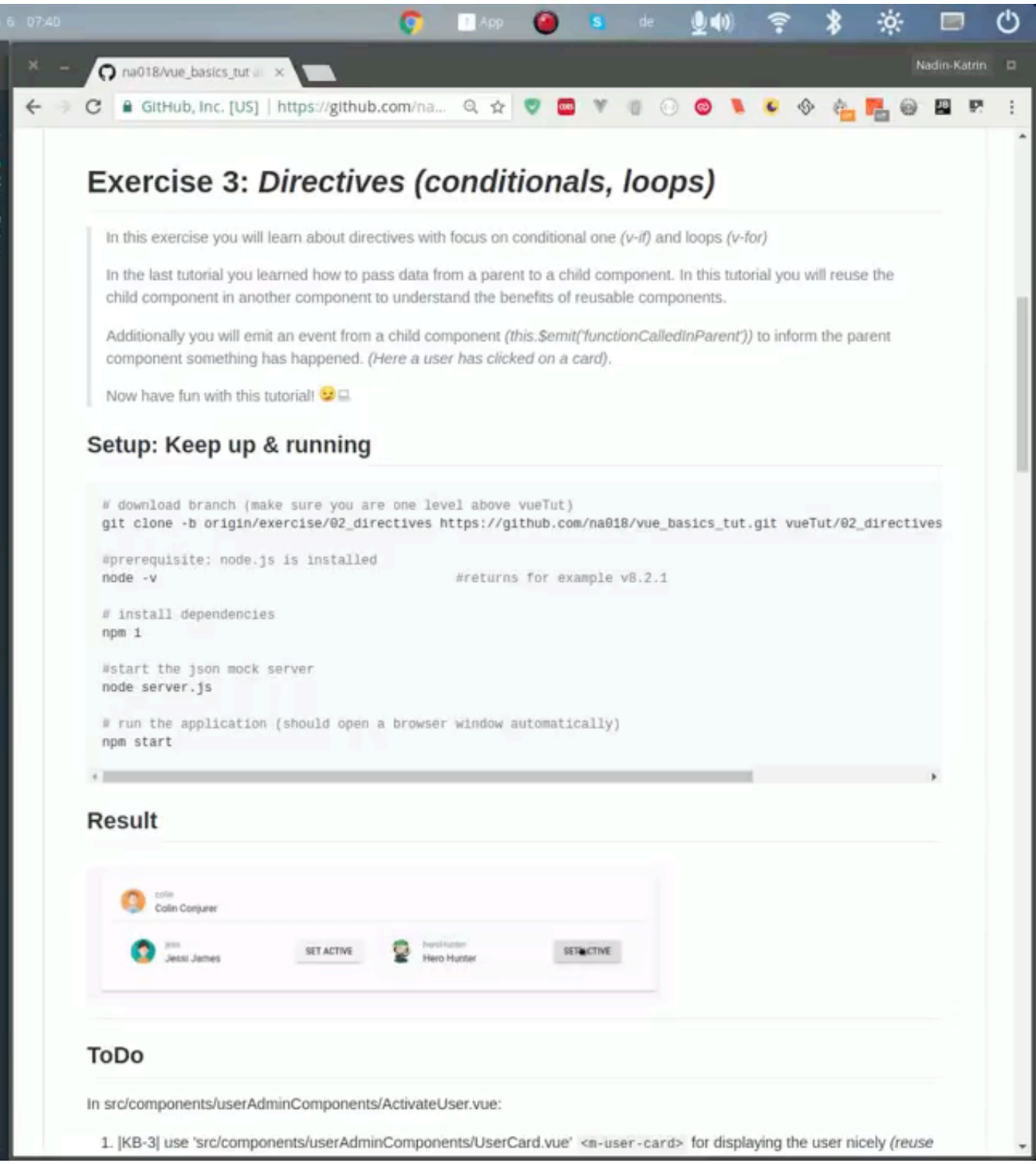
Recap: kanban components



https://youtu.be/UEdEMPrsQ1U

Recap: kanban loops

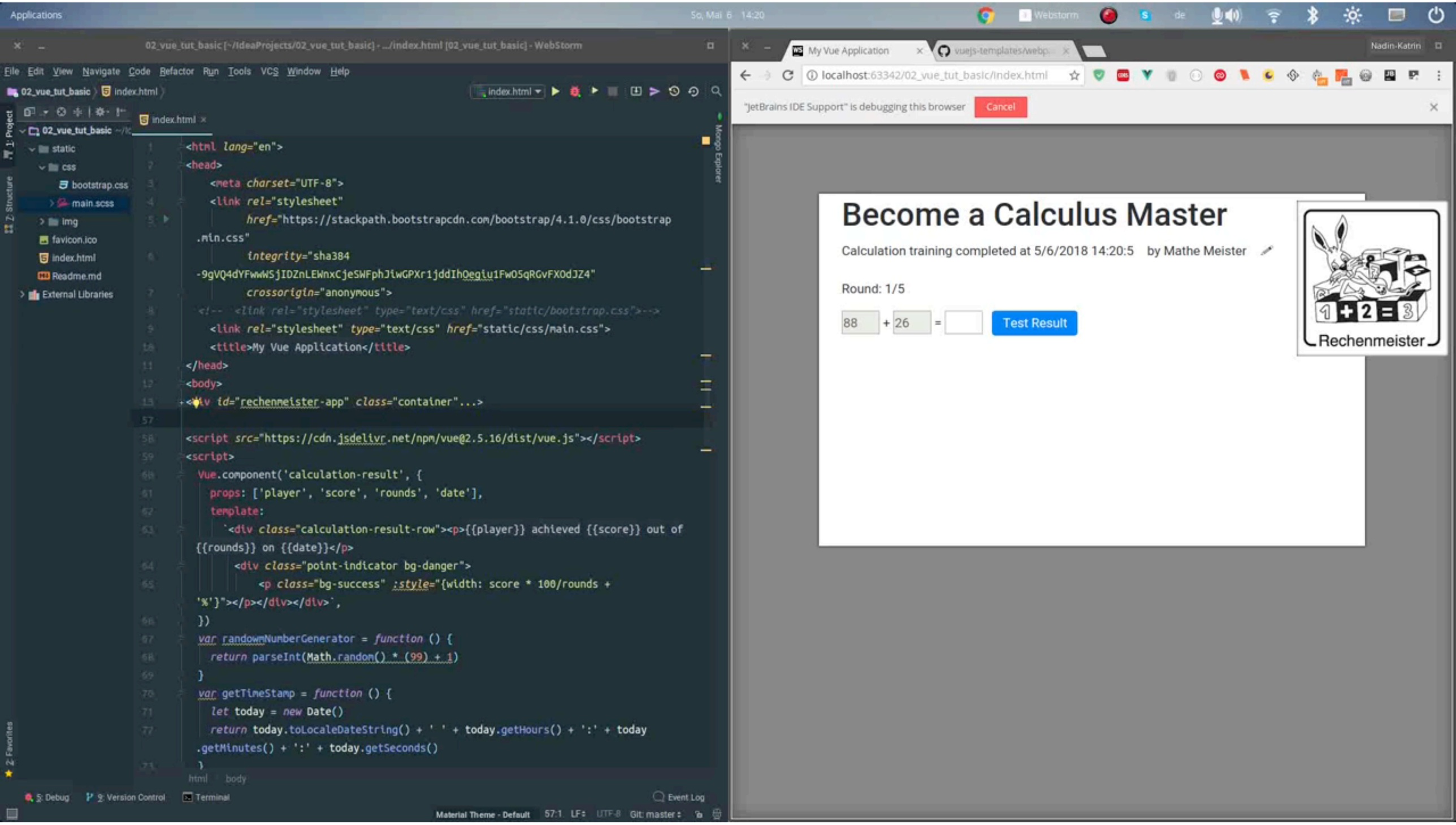




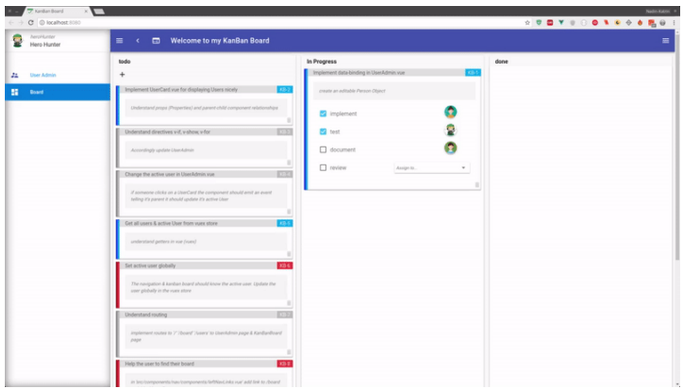
https://youtu.be/4DCzkokv4uE

Table of Contents

The Vue CLI



https://youtu.be/zCHAYQ-YgT8



Thanks for your Attention

Nadin-Katrin Apel, Alexander Schübl, David Bochan

You should now understand the basics of Vue.js