**Section 35 – Improving Your App With Filters And Mixins**

Chapter 503 – Chapter 512

1. Give some filters to our data, only control filters on the file.

<template>

    <div class="container">

        <div class="row">

            <div class="col-xs-12 col-sm-8 col-sm-offset-2 col-md-6 col-md-offset-3">

                <h1>Filters & Mixins</h1>

                <p>{{ text | toUppercase }}</p>

            </div>

        </div>

    </div>

</template>

<script>

export default {

    data() {

        return {

            text: 'Hello there !'

        }

    },

    filters: {

        toUppercase(value) {

            return value.toUppercase();

        }

    }

}

</script>

1. Global filters.

The example below the text will change into uppercase and then will change into lowercase.

<p>{{ text | toUppercase | to-lowecase }}</p>

1. Filter Some Data

<template>

    <div class="container">

        <div class="row">

            <div class="col-xs-12 col-sm-8 col-sm-offset-2 col-md-6 col-md-offset-3">

                <h1>Filters & Mixins</h1>

                <p>{{ text | toUppercase | to-lowecase }}</p>

                <hr>

                <input v-model="filterText">

                <ul>

                    <li v-for="fruit in filteredFruits">{{ fruit }}</li>

                </ul>

            </div>

        </div>

    </div>

</template>

<script>

export default {

    data() {

        return {

            text: 'Hello there !',

            fruits: ['Apple', 'Banana', 'Mango', 'Melon'],

            filterText: ''

        }

    },

    filters: {

        toUppercase(value) {

            return value.toUppercase();

        }

    },

    computed: {

        filteredFruits() {

            return this.fruits.filter((element) => {

                return element.match(this.filterText)

            });

        }

    }

}

</script>

1. Import Mixins

List.vue

<template>

    <div>

        <h1>Filters & Mixins</h1>

        <input v-model="filterText">

        <ul>

            <li v-for="fruit in filteredFruits">{{ fruit }}</li>

        </ul>

    </div>

</template>

<script>

import { fruitMixin } from './fruitMixin';

export default {

    mixins: [fruitMixin]

}

</script>

<style>

</style>

fruitMixin.js

const fruitMixin = {

    data() {

        return {

            fruits: ['Apple', 'Banana', 'Mango', 'Melon'],

            filterText: ''

        }

    },

    computed: {

        filteredFruits() {

            return this.fruits.filter((element) => {

                return element.match(this.filterText)

            });

        }

    }

}

1. Process Created based on Mixins are

Global Mixin => Mixins File (fruitMixins.js) => File (Data such as App.vue, List.vue)



1. Global Mixins

If we have 1 mixins, 2 files call this mixins, and the first file add one fruit into the mixins fruit array, such as berries, so the array will contain berries at the first file, but for the second file, the array will not contain berries.

1. Class

App.vue

<template>

    <div class="container">

        <div class="row">

            <div class="col-xs-12 col-sm-8 col-sm-offset-2 col-md-6 col-md-offset-3">

                <h1>Filters & Mixins</h1>

                <p>{{ text | toUppercase | to-lowecase }}</p>

                <hr>

                <button @click="fruits.push('Berries')">Add New Item</button>

                <input v-model="filterText">

                <ul>

                    <li v-for="fruit in filteredFruits">{{ fruit }}</li>

                </ul>

                <hr>

                <app-list></app-list>

            </div>

        </div>

    </div>

</template>

<script>

import List from './List.vue';

import { fruitMixin } from './fruitMixin';

export default {

    mixins: [fruitMixin],

    data() {

        return {

            text: 'Hello there !',

            fruits: ['Apple', 'Banana', 'Mango', 'Melon'],

            filterText: ''

        }

    },

    filters: {

        toUppercase(value) {

            return value.toUppercase();

        }

    },

    computed: {

        filteredFruits() {

            return this.fruits.filter((element) => {

                return element.match(this.filterText)

            });

        }

    },

    components: {

        appList: List

    }

}

</script>

<style>

</style>

List.vue

<template>

    <div>

        <h1>Filters & Mixins</h1>

        <input v-model="filterText">

        <ul>

            <li v-for="fruit in filteredFruits">{{ fruit }}</li>

        </ul>

    </div>

</template>

<script>

import { fruitMixin } from './fruitMixin';

export default {

    mixins: [fruitMixin],

    created() {

        console.log('Inside List Hook');

    }

}

</script>

<style>

</style>

fruitMixin.js

const fruitMixin = {

    data() {

        return {

            fruits: ['Apple', 'Banana', 'Mango', 'Melon'],

            filterText: ''

        }

    },

    computed: {

        filteredFruits() {

            return this.fruits.filter((element) => {

                return element.match(this.filterText);

            });

        }

    },

    created() {

        console.log('Created');

    }

}

main.js

import Vue from 'vue'

import App from './App.vue'

Vue.filter('to-lowercase', function() {

  return value.toLowerCase();

});

Vue.mixin({

  created() {

    console.log('Global Mixin - Created Hook');

  }

});

new Vue({

  el: '#app',

  render: h => h(App)

});