**Introdução aos Componentes**

**Aula 105 – Revisão sobre componentes**

1. Vamos criar uma pasta chamada componentes e dentro dela um arquivo chamado revisão.html.
2. Vamos usar a forma antiga referenciando o vue diretamente no arquivo e depois iremos utilizar o vue-create ainda neste capítulo.
3. <meta *charset*="UTF-8">
4. <script *src*="https://cdn.jsdelivr.net/npm/vue/dist/vue.js"></script>
5. <div *id*="app">
6. <span>{{contador}}</span>
7. <button *@click*="contador++">+</button>
8. <button *@click*="contador--">-</button>
9. </div>
10. <script>
11. **new** Vue({
12. el:'#app',
13. data: {
14. contador: 0
15. }
16. })
17. </script>
18. Desta forma não é possível duplicar as div’s, temos que criar compontentes.
19. Podemos criar utilizando a forma glogal. Ficando assim:
20. <meta *charset*="UTF-8">
21. <script *src*="https://cdn.jsdelivr.net/npm/vue/dist/vue.js"></script>
22. <div *id*="app">
23. <contador></contador>
24. </div>
25. <script>
26. Vue.component('contador',{
27. template:`
28. <div>
29. <span>{{contador}}</span>
30. <button @click="contador++">+</button>
31. <button @click="contador--">-</button>
32. </div>
33. `,
34. data() {
35. return{
36. contador: 0
37. }
38. }
39. })
40. **new** Vue({
41. el:'#app',
43. })
44. </script>
45. Agora para utilizarmos mais de um componente basta inseri-los dentro da div app.
46. <div *id*="app">
47. <contador></contador>
48. <contador></contador>
49. <contador></contador>
50. </div>

**Aula 106 -Registrando componentes (Local e global)**

1. Vamos mostrar outra forma de registrar um componente, para isso vamos criar o arquivo compontente-registro.html.
2. Vamos duplicar a div app, criando uma div com app1 e outra app2, vamos também duplicar a instância vue apontando para app1 e outra para app2.
3. <meta *charset*="UTF-8">
4. <script *src*="https://cdn.jsdelivr.net/npm/vue/dist/vue.js"></script>
5. <div *id*="app1">
6. <contador></contador>
7. <contador></contador>
8. <contador></contador>
9. </div>
10. <hr>
11. <div *id*="app2">
12. <contador></contador>
13. <contador></contador>
14. <contador></contador>
15. </div>
16. <script>
17. Vue.component('contador',{
18. template:`
19. <div>
20. <span>{{contador}}</span>
21. <button @click="contador++">+</button>
22. <button @click="contador--">-</button>
23. </div>
24. `,
25. data() {
26. return{
27. contador: 0
28. }
29. }
30. })
31. **new** Vue({
32. el:'#app1',
34. })
35. **new** Vue({
36. el:'#app2',
38. })
39. </script>
40. Isso só é possível devido termos criado o componente de forma global.

**Criando componente Local**

1. Vamos utilizar forma local de criação de componente, para isso vamos apagar o vue.componente e criar uma constante do tipo objeto chamada contador e vamos passar todo o conteúdo do componente.
2. Agora dentro da instância do vue app1 vamos criar um atributo chamado components, dentro de components vamos criar uma variável chamada contador1 que irá receber a variável contador. Agora a primeira instância app1 irá funcionar.
3. Vamos repetir o mesmo procedimento na instância app2 porém com o component chamado contador2.
4. <meta *charset*="UTF-8">
5. <script *src*="https://cdn.jsdelivr.net/npm/vue/dist/vue.js"></script>
6. <div *id*="app1">
7. <contador1></contador1>
8. <contador1></contador1>
9. <contador1></contador1>
10. </div>
11. <hr>
12. <div *id*="app2">
13. <contador2></contador2>
14. <contador2></contador2>
15. <contador2></contador2>
16. </div>
17. <script>
18. const contador = {
19. template:`
20. <div>
21. <span>{{contador}}</span>
22. <button @click="contador++">+</button>
23. <button @click="contador--">-</button>
24. </div>
25. `,
26. data() {
27. return{
28. contador: 0
29. }
30. }
31. }
33. **new** Vue({
34. el:'#app1',
35. components:{
36. contador1: contador
37. }
38. })
39. **new** Vue({
40. el:'#app2',
41. components:{
42. contador2: contador
43. }
45. })
46. </script>