

1. Given a MoviesList component, which of the following options allows `<movies-list v-bind:movies="listOfMovies">`?

Show Results

18/18 Students Answered

A `export default {  
 name: 'movies-list',  
 data() {  
 return {  
 movies: []  
 }  
 }  
}`



B `export default {  
 name: 'movies-list',  
 computed: {  
 movies() {  
 return [];  
 }  
 }  
}`



C `export default {  
 name: 'movies-list',  
 input() {  
 return {  
 movies: []  
 }  
 }  
}`



D `export default {  
 name: 'movies-list',  
 props: ['movies']  
}`



['movies']



How can the MoviesList component have access to the List of movies from the parent component?

props!

2. Given a MoviesList component, which of the following options allows `<movies-list search="Club">?`

Show Results

18/18 Students Answered

**A** `export default {  
 name: 'movies-list',  
 data() {  
 return {  
 movies: [],  
 search: ""  
 }  
 }  
}`

**B** `export default {  
 name: 'movies-list',  
 computed: {  
 movies() {  
 return [];  
 }  
 },  
 data() {  
 return {  
 search: ""  
 }  
 }  
}`

**C** `export default {  
 name: 'movies-list',  
 input() {  
 return {  
 movies: [],  
 search: ""  
 }  
 }  
}`

**D** `export default {  
 name: 'movies-list',  
 data() {  
 return {  
 movies: []  
 }  
 }  
 props: ['search']  
}`



4. Given MoviesList.vue, as declared below, which of the following do you need to do to add it into another component? (select all that apply):

```
export default {  
  name: 'movies-list'  
}
```

Show Results

18/18 Students Answered

A `<script>  
import MoviesList from './components/MoviesList.vue';  
</script>` ✓

B `<script>  
export default {  
 components: {  
 MoviesList  
 }  
};  
</script>` ✓

C `<template>  
<Movies></Movies>  
</template>` ✗

D `<script>  
import default {  
 modules: [  
 './MoviesList.vue'  
 ]  
};  
</script>` ✗

E `<template>  
<movies-list></movies-list>  
</template>` ✓

3. Why should your components follow the Single Responsibility Principle (SRP)?

Show Results

18/18 Students Answered

- A You don't want your files to get too large.
- B Computers can only do one thing at a time.
- C It makes it easier to maintain, test, and reuse your components. ✓
- D Because developers are paid by lines of code written.

6. How would a component access an object named pets kept in a Vuex datastore?

Show Results

18/18 Students Answered

- A this.\$store.data.pets ✗
- B this.\$data.store.pets ✗
- C this.\$store.state.pets ✓
- D this.\$vuex.data.pets ✗
- E this.\$store.pets ✗

## 7. How do you call a mutation called ADD\_REVIEW?

Show Results

18/18 Students Answered

- A `this.$store.ADD_REVIEW();`
- B `this.$store.commit('ADD_REVIEW');` ✓
- C `this.$store.mutation('ADD_REVIEW');`
- D `this.$store.mutation.ADD_REVIEW();`
- E `this.$store.trigger('ADD_REVIEW');`

## 5. Given the component below, how would you pass an object named productReviews into it from a parent component?

```
<template>
<div v-for="review in reviews" v-bind:key="review.id"></div>
</template>
```

```
<script>
export default {
  name: 'book-reviews',
  props: ['reviews']
};
</script>
```

Show Results

18/18 Students Answered

- A `<book-reviews v-prop="productReviews"></book-reviews>` ✓
- B `<product-reviews v-bind:prop="book-reviews"></product-reviews>` ✗
- C `<book-reviews v-bind:reviews="productReviews"></book-reviews>` ✓
- D `<product-reviews v-prop="book-reviews"></product-reviews>` ✗

```
<div class="reviews">
  <review-display
    v-for="review in filteredReviews"
    v-bind:key="review.title" ← review.title
    v-bind:review="review"
  />
</div>
</template>
```

for (let i = 0; i < filteredReviews.length; i++) {  
 let review = filteredReviews[i];

```
<template>
  <div class="reviews">
    <review-display
      v-for="review in filteredReviews"
      v-bind:key="review.title"
      v-bind:review="review" v-bind:review="singleReview"
    />
  </div>
</template>

<script>
import ReviewDisplay from "../ReviewDisplay";

export default {
  name: "review-list",
  components: {
    ReviewDisplay
  },
  data() {
    return {
      reviews: [
        {
          id: 1,
          title: "This is a great product",
          author: "John Doe",
          date: "2020-01-01",
          favorite: false
        },
        {
          id: 2,
          title: "This is a terrible product",
          author: "Jane Doe",
          date: "2020-01-01",
          favorite: false
        },
        {
          id: 3,
          title: "This is an average product",
          author: "John Doe",
          date: "2020-01-01",
          favorite: false
        }
      ]
    };
  },
  methods: {
    onFavoritedChange(review) {
      this.$store.commit("FLIP_FAVORITED", review);
    }
  }
};
```

```
ReviewList.vue > {} "ReviewList.vue" > template > div.reviews > review-display
1 <template> PARENT COMPONENT
2 <div class="reviews">
3 <review-display
4 v-for="review in filteredReviews"
5 v-bind:key="review.title" "review.id" -- made up!!!
6 v-bind:review="review"
7 />
8 </div>
9 </template>
10
11 <script>
12 import ReviewDisplay from "../ReviewDisplay";
13
14 export default {
15   name: "review-list",
16   components: {
17     ReviewDisplay,
18   },
19   data() {
20     return {
21       reviews: [
22         {
23           id: 1,
24           title: "This is a great product",
25           author: "John Doe",
26           date: "2020-01-01",
27           favorite: false
28         },
29         {
30           id: 2,
31           title: "This is a terrible product",
32           author: "Jane Doe",
33           date: "2020-01-01",
34           favorite: false
35         },
36         {
37           id: 3,
38           title: "This is an average product",
39           author: "John Doe",
40           date: "2020-01-01",
41           favorite: false
42         }
43       ]
44     };
45   },
46   methods: {
47     onFavoritedChange(review) {
48       this.$store.commit("FLIP_FAVORITED", review);
49     }
50   }
51 };
52 </script>
53 </div>
54 </template>
55 </div>
56 </template>
57 </div>
58 </template>
59 </div>
60 </template>
61 </div>
62 </template>
63 </div>
64 </template>
65 </div>
66 </template>
67 </div>
68 </template>
69 </div>
70 </template>
71 </div>
72 </template>
73 </div>
74 </template>
75 </div>
76 </template>
77 </div>
78 </template>
79 </div>
80 </template>
81 </div>
82 </template>
83 </div>
84 </template>
85 </div>
86 </template>
87 </div>
88 </template>
89 </div>
90 </template>
91 </div>
92 </template>
93 </div>
94 </template>
95 </div>
96 </template>
97 </div>
98 </template>
99 </div>
100 </template>
```

when you have a v-for (loop)  
vue requires you to have a key  
the key must be unique