

NAME- RISHIT TIWARI
ROLL NO- 22CS3050

WEB TECH LAB-09

Q1.

```
App.vue x style.css x script.js x
1 <template>
2   <div>
3     <h1>Currency Converter</h1>
4     <div>
5       <label>
6         Amount:
7         <input type="number" v-model="amount" @input="handleAmountChange" />
8       </label>
9     </div>
10    <div>
11      <label>
12        From Currency:
13        <select v-model="fromCurrency" @change="handleFromCurrencyChange">
14          <option value="INR">INR</option>
15          <option value="USD">USD</option>
16        </select>
17      </label>
18    </div>
19    <div>
20      <label>
21        To Currency:
22        <select v-model="toCurrency" @change="handleToCurrencyChange">
23          <option value="USD">USD</option>
24          <option value="INR">INR</option>
25        </select>
26      </label>
27    </div>
28    <div>
29      <button @click="convertCurrency">Convert</button>
30    </div>
31  </div>
32  <div>
33    Converted Amount: {{ convertedAmount }} {{ toCurrency }}
34  </div>
35</template>
36
37<script>
38export default {
39  data() {
40    return {
41      amount: '',
42      fromCurrency: 'INR',
43      toCurrency: 'USD',
44      exchangeRateINRtoUSD: 0.012,
45      exchangeRateUSDtoINR: 82.78
46    };
47  },
48  computed: {
49    convertedAmount() {
50      if (this.fromCurrency === 'INR' && this.toCurrency === 'USD') {
51        return (this.amount * this.exchangeRateINRtoUSD).toFixed(2);
52      } else if (this.fromCurrency === 'USD' && this.toCurrency === 'INR') {
53        return (this.amount * this.exchangeRateUSDtoINR).toFixed(2);
54      } else {
55        return 'Invalid conversion';
56      }
57    }
58  }
59}
```

```
App.vue x style.css x script.js x
54     } else {
55       return 'Invalid conversion';
56     }
57   }
58 },
59 methods: {
60   handleAmountChange(event) {
61     this.amount = event.target.value;
62   },
63   handleFromCurrencyChange(event) {
64     this.fromCurrency = event.target.value;
65   },
66   handleToCurrencyChange(event) {
67     this.toCurrency = event.target.value;
68   },
69   convertCurrency() {
70   }
71 }
72 };
73 </script>
74
```

Currency Converter

Amount:

From Currency:

To Currency:

Converted Amount: 0.000 USD

Currency Converter

Amount:

From Currency:

To Currency:

Converted Amount: 0.012 USD

Currency Converter

Amount:

From Currency: USD ▼

To Currency: INR ▼

Converted Amount: 82.910 INR

Currency Converter

Amount:

From Currency: USD ▼

To Currency: INR ▼

Converted Amount: 829.100 INR

Q2.

```
App.vue x style.css x script.js x
1 <template>
2 <div>
3 <h1>Stopwatch</h1>
4 <div>
5 <p>{{ formatTime(time) }}</p>
6 </div>
7 <div>
8 <button @click="startStopwatch" v-if="!isRunning">Start</button>
9 <button @click="pauseStopwatch" v-else>Pause</button>
10 <button @click="resetStopwatch">Reset</button>
11 </div>
12 </div>
13 </template>
14
15 <script>
16 export default {
17   data() {
18     return {
19       time: 0,
20       isRunning: false,
21       intervalId: null
22     };
23   },
24   methods: {
25     startStopwatch() {
26       this.isRunning = true;
27       this.intervalId = setInterval(() => {
28         this.time++;
29       }, 1000);
30     },
31     pauseStopwatch() {
32       this.isRunning = false;
33       clearInterval(this.intervalId);
34     },
35     resetStopwatch() {
36       this.isRunning = false;
37       this.time = 0;
38       clearInterval(this.intervalId);
39     },
40     formatTime(seconds) {
41       const hours = Math.floor(seconds / 3600);
42       const minutes = Math.floor((seconds % 3600) / 60);
43       const remainingSeconds = seconds % 60;
44       return `${hours.toString().padStart(2, '0')}:${minutes
45         .toString()
46         .padStart(2, '0')}:${remainingSeconds.toString().padStart(2, '0')}`;
47     }
48   },
49   beforeDestroy() {
50     clearInterval(this.intervalId);
51   }
52 };
53 </script>
54
```

Stopwatch

00:00:00

Start Reset

Stopwatch

00:00:05

Pause Reset

Stopwatch

00:00:00

Start Reset

Q3.

```
App.vue x style.css x script.js x
1 <template>
2   <div class="messaging-app">
3     <h1>Messaging App</h1>
4     <div class="container">
5       <div class="row">
6         <div class="col-md-3">
7           <h2>Conversations</h2>
8           <ul class="list-group">
9             <li
10               v-for="conversation in conversations"
11               :key="conversation.id"
12               class="list-group-item"
13               @click="handleConversationClick(conversation.id)"
14               :class="{ active: conversation.id === selectedConversation }"
15             >
16               {{ conversation.name }}
17             </li>
18           </ul>
19         </div>
20
21         <div class="col-md-9">
22           <div v-if="selectedConversation" class="chat-box">
23             <h2>{{ `Chat with ${conversations.find((conv) => conv.id === selectedConversation).name}` }}</h2>
24             <div
25               v-for="(message, index) in conversations.find((conv) => conv.id === selectedConversation).messages"
26               :key="index"
27               class="message"
28             >
29               <div class="message-bubble">
30
31                 <p>{{ message.text }}</p>
32                 <span class="message-timestamp">{{ message.timestamp }}</span>
33               </div>
34             </div>
35             <div>
36               <div class="form-control" v-model="newMessage"></div>
37               <button class="btn btn-primary mt-2" @click="handleSendMessage">Send</button>
38             </div>
39           </div>
40         </div>
41       </div>
42     </div>
43   </template>
44
45   <script>
46   export default {
47     data() {
48       return {
49         conversations: [
50           { id: 1, name: 'Family', messages: [] },
51           { id: 2, name: 'Friends', messages: [] }
52         ],
53         selectedConversation: null,
54         newMessage: ''
55       };
56     },
57     methods: {
58       handleConversationClick(conversationId) {
59         this.selectedConversation = conversationId;
```

```
App.vue x style.css x script.js x
56     handleConversationClick(conversationId) {
57       this.selectedConversation = conversationId;
58     },
59     handleSendMessage() {
60       if (this.newMessage.trim() !== '' && this.selectedConversation) {
61         const conversationIndex = this.conversations.findIndex(conv => conv.id === this.selectedConversation);
62         this.conversations[conversationIndex].messages.unshift({
63           text: this.newMessage,
64           timestamp: new Date().toLocaleTimeString()
65         });
66         this.newMessage = '';
67       }
68     }
69   }
70 };
71 </script>
72
73 <style scoped>
74   .messaging-app {
75     background-color: #f8f9fa;
76     padding: 20px;
77   }
78
79   .container {
80     background-color: #fff;
81     border-radius: 10px;
82     padding: 20px;
83   }
```

Messaging App

Conversations

- Family
- Friends

Messaging App

Conversations

- Family
- Friends

Chat with Family

Send

Messaging App

Conversations

- Family
- Friends

Chat with Family

pranam ghar pe sabko
4:38:03 PM

Send

Messaging App

Conversations

- Family
- Friends

Chat with Family

main sahi hoon
4:38:21 PM

pranam ghar pe sabko
4:38:03 PM

Send

Messaging App

Conversations

- Family
- Friends

Chat with Friends

aur bhaiyon
4:38:41 PM

Send