

LAB – 9

NAME – RAJ

ROLL NO – 22CS3045

T1.

```
<!DOCTYPE html>

<html lang="en">

<head>

  <meta charset="UTF-8">

  <meta name="viewport" content="width=device-width, initial-scale=1.0">

  <title>Currency Converter</title>

  <script src="https://cdn.jsdelivr.net/npm/vue@2"></script>

</head>

<body>

  <div id="app">

    <h1>Currency Converter</h1>

    <div>

      <label for="amount">Enter Amount:</label>

      <input type="number" id="amount" v-model="amount" @input="convertCurrency">

    </div>

    <div>

      <label for="from">From:</label>

      <select id="from" v-model="fromCurrency" @change="convertCurrency">

        <option value="USD">USD</option>

        <option value="EUR">EUR</option>

        <option value="GBP">GBP</option>

      </select>

    </div>

    <div>

      <label for="to">To:</label>
```

```

    <select id="to" v-model="toCurrency" @change="convertCurrency">
      <option value="USD">USD</option>
      <option value="EUR">EUR</option>
      <option value="GBP">GBP</option>
    </select>
  </div>
  <div>
    <p>{{ amount }} {{ fromCurrency }} equals {{ convertedAmount }} {{ toCurrency }}</p>
  </div>
</div>

```

```

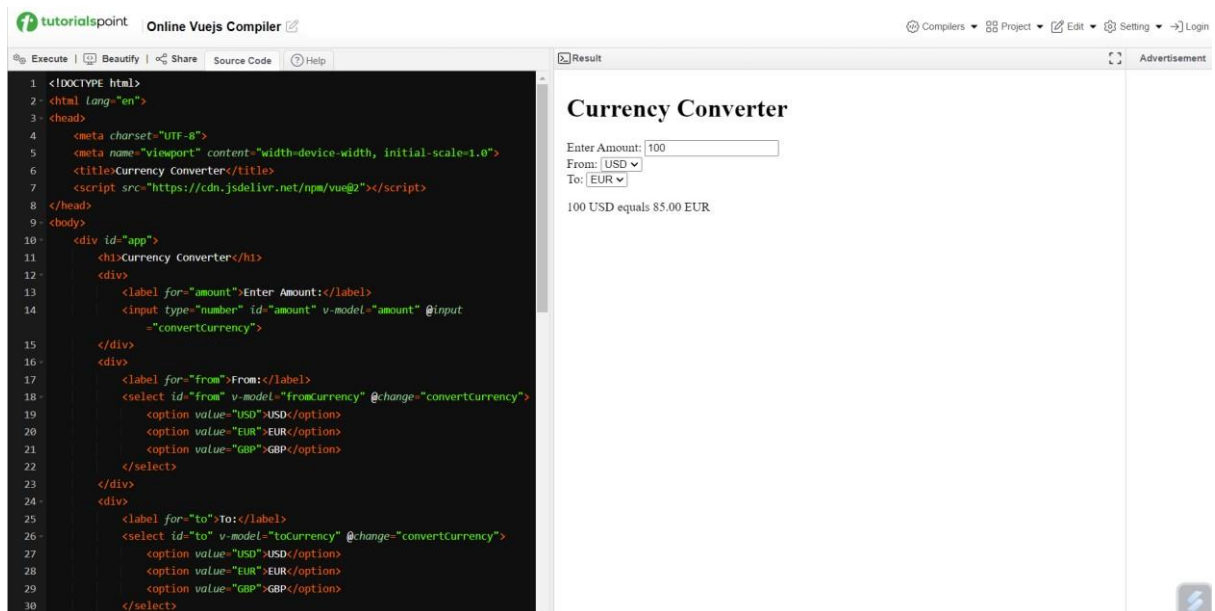
<script>
new Vue({
  el: '#app',    data:
{
  amount: 0,
fromCurrency: 'USD',
toCurrency: 'USD',
exchangeRates: {
  USD: {
    EUR: 0.85,
    GBP: 0.73
  },
  EUR: {
    USD: 1.18,
    GBP: 0.86
  },
  GBP: {
    USD: 1.38,
    EUR: 1.16
  }
},

```

```

        convertedAmount: 0
    },
    methods: {
        convertCurrency: function () {
            const rate
            = this.exchangeRates[this.fromCurrency][this.toCurrency];
            this.convertedAmount = (this.amount * rate).toFixed(2);
        }
    }
});
</script>
</body>
</html>

```



The screenshot shows the Online Vue.js Compiler interface. On the left, the source code is displayed in a dark-themed editor. On the right, the rendered output is shown in a light-themed browser window.

Source Code:

```

1 <!DOCTYPE html>
2 <html lang="en">
3 <head>
4   <meta charset="UTF-8">
5   <meta name="viewport" content="width=device-width, initial-scale=1.0">
6   <title>Currency Converter</title>
7   <script src="https://cdn.jsdelivr.net/npm/vue@2"></script>
8 </head>
9 <body>
10  <div id="app">
11    <h1>Currency Converter</h1>
12    <div>
13      <label for="amount">Enter Amount:</label>
14      <input type="number" id="amount" v-model="amount" @input
15        ="convertCurrency">
16    </div>
17    <div>
18      <label for="from">From:</label>
19      <select id="from" v-model="fromCurrency" @change="convertCurrency">
20        <option value="USD">USD</option>
21        <option value="EUR">EUR</option>
22        <option value="GBP">GBP</option>
23      </select>
24    </div>
25    <div>
26      <label for="to">To:</label>
27      <select id="to" v-model="toCurrency" @change="convertCurrency">
28        <option value="USD">USD</option>
29        <option value="EUR">EUR</option>
30        <option value="GBP">GBP</option>
31      </select>

```

Rendered Output:

Currency Converter

Enter Amount:

From: USD ▼

To: EUR ▼

100 USD equals 85.00 EUR

T2.

```
<!DOCTYPE html>
```

```
<html lang="en">
```

```
<head>
```

```
  <meta charset="UTF-8">
```

```

<meta name="viewport" content="width=device-width, initial-scale=1.0">
<title>Stopwatch</title>
<script src="https://cdn.jsdelivr.net/npm/vue@2"></script>
</head>
<body>
  <div id="app">
    <h1>Stopwatch</h1>
    <div>
      <p>{{ formattedTime }}</p>
    </div>
    <div>
      <button @click="startStopwatch" v-if="!isRunning">Start</button>
      <button @click="pauseStopwatch" v-if="isRunning">Pause</button>
      <button @click="resetStopwatch">Reset</button>
    </div>
  </div>

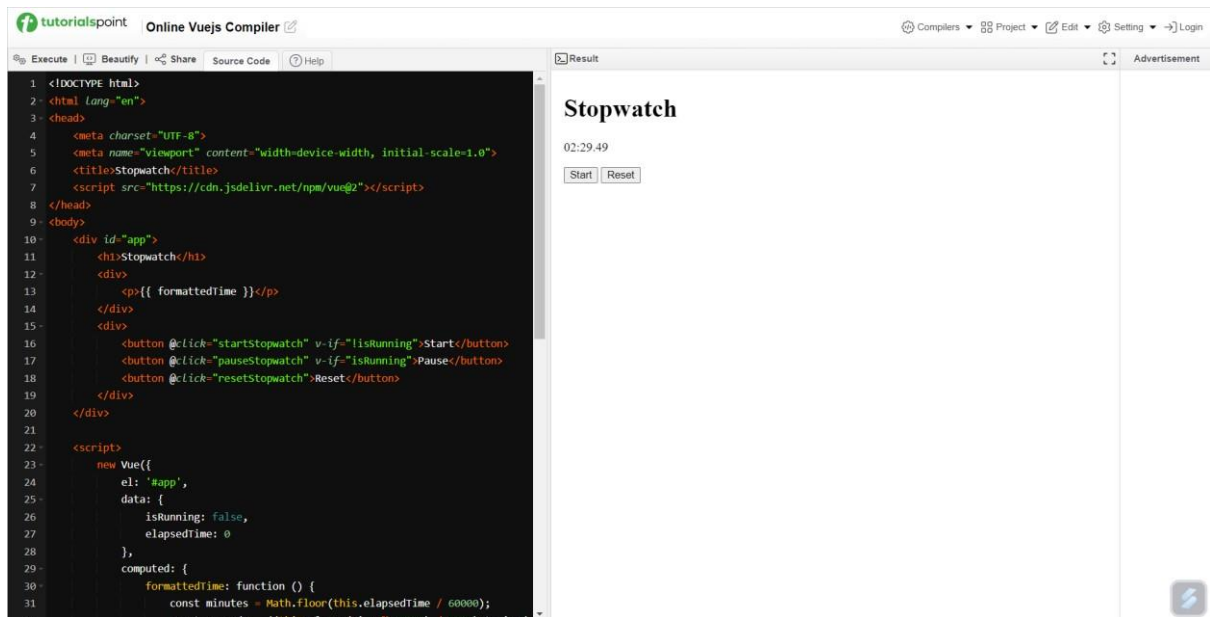
  <script>
new Vue({
  el: '#app',
  data: {
    isRunning: false,
    elapsedTime: 0
  },
  computed: {
    formattedTime: function () {
      const minutes =
Math.floor(this.elapsedTime /
60000);          const seconds =
((this.elapsedTime % 60000) /
1000).toFixed(2);          return

```

```

`${minutes < 10 ? '0' :
"}${minutes}:${seconds < 10 ? '0' :
"}${seconds}`;
    }
    },
    methods: {
startStopwatch: function () {
    if (!this.isRunning) {        this.isRunning = true;
this.startTime = Date.now() - this.elapsedTime;
this.timer = setInterval(this.updateElapsedTime, 10);
    }
    },
    pauseStopwatch: function () {
if (this.isRunning) {
this.isRunning = false;
clearInterval(this.timer);
    }
    },
    resetStopwatch: function () {
this.isRunning = false;
clearInterval(this.timer);
this.elapsedTime = 0;
    },
    updateElapsedTime: function () {
this.elapsedTime = Date.now() - this.startTime;
    }
    }
    });
</script>
</body>
</html>

```



T3.

```
<!DOCTYPE html>
```

```
<html lang="en">
```

```
<head>
```

```
  <meta charset="UTF-8">
```

```
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
```

```
  <title>Messaging App</title>
```

```
  <script src="https://cdn.jsdelivr.net/npm/vue@2"></script>
```

```
  <!-- Firebase -->
```

```
  <script src="https://www.gstatic.com/firebasejs/9.1.3/firebase-app.js"></script>
```

```
  <script src="https://www.gstatic.com/firebasejs/9.1.3/firebase-database.js"></script>
```

```
</head>
```

```
<body>
```

```
  <div id="app">
```

```
    <h1>Messaging App</h1>
```

```
    <div class="conversation-list">
```

```
      <ul>
```

```

        <li v-for="(conversation, index) in conversations" :key="index"
@click="selectConversation(conversation)">

            USER-01

        </li>

        <li v-for="(conversation, index) in conversations" :key="index"
@click="selectConversation(conversation)">

            USER-02

        </li>

    </ul>

</div>

<div v-if="selectedConversation">

    <h2>Conversations</h2>

    <div class="message-list">

        <div v-for="(message, index) in messages" :key="index">

            <p>USER-01:HELLO, HOW ARE YOU?</p>

            <p>USER-02:I AM FINE. WHAT ABOUT YOU?</p>

        </div>

    </div>

    <div class="message-input">

        <input type="text" v-model="newMessageText" placeholder="Type your message...">

    <button @click="sendMessage">Send</button>

    </div>

</div>

</div>

<script>

    // Initialize Firebase

const firebaseConfig = {

    // Your Firebase configuration

};

firebase.initializeApp(firebaseConfig);

```

```

new Vue({
  el: '#app',
  data: {
    conversations: [
      { id: 1, name: 'Friend 1' },
      { id: 2, name: 'Friend 2' },
      // Add more conversations as needed
    ],
    selectedConversation: null,
    messages: [],
    newMessageText: ''
  },
  mounted() {
    // Load initial conversation or set selectedConversation
    // Here, you can also handle authentication
  },
  methods: {
    selectConversation(conversation) {
      this.selectedConversation = conversation;
      this.loadMessages(conversation.id);
    },
    loadMessages(conversationId) {
      // Load messages from Firebase for the selected conversationId
      firebase.database().ref('messages/' + conversationId).on('value', (snapshot) => {
        this.messages = snapshot.val() || [];
      });
    },
    sendMessage() {
      if (this.newMessageText.trim() === '') return;

      // Push the new message to Firebase
    }
  }
});

```



```

        firebase.database().ref('messages/' + this.selectedConversation.id).push({
sender: 'You', // You can add the sender's username here        text:
this.newMessageText

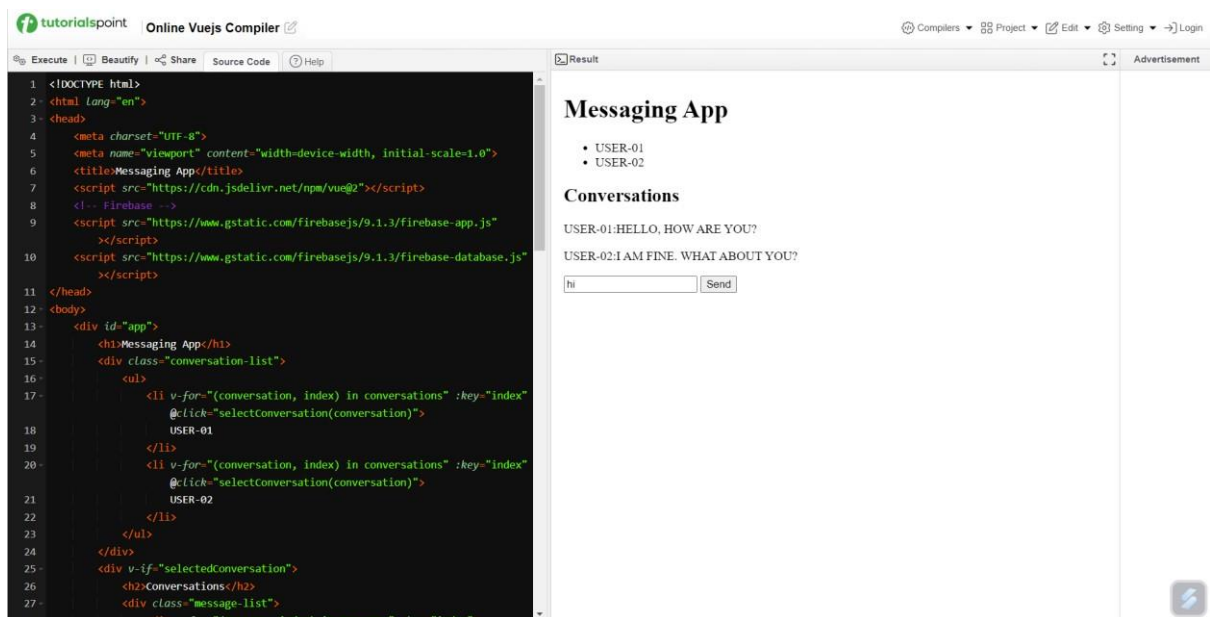
        });

        // Clear the input field after sending the message
this.newMessageText = '';

    }
}

});
</script>
</body>
</html>

```



The screenshot shows the Online VueJS Compiler interface. On the left, the source code is displayed, and on the right, the rendered output is shown.

Source Code:

```

1 <!DOCTYPE html>
2 <html Lang="en">
3 <head>
4   <meta charset="UTF-8">
5   <meta name="viewport" content="width=device-width, initial-scale=1.0">
6   <title>Messaging App</title>
7   <script src="https://cdn.jsdelivr.net/npm/vue@2"></script>
8   <!-- Firebase -->
9   <script src="https://www.gstatic.com/firebasejs/9.1.3/firebase-app.js"
10  ></script>
11   <script src="https://www.gstatic.com/firebasejs/9.1.3/firebase-database.js"
12  ></script>
13 </head>
14 <body>
15   <div id="app">
16     <h1>Messaging App</h1>
17     <div class="conversation-list">
18       <ul>
19         <li v-for="(conversation, index) in conversations" :key="index"
20           @click="selectConversation(conversation)">
21           USER-01
22         </li>
23         <li v-for="(conversation, index) in conversations" :key="index"
24           @click="selectConversation(conversation)">
25           USER-02
26         </li>
27       </ul>
28     </div>
29     <div v-if="selectedConversation">
30       <h2>Conversations</h2>
31       <div class="message-list">
32         <div v-for="(message, index) in messages" :key="index">

```

Rendered Output:

Messaging App

- USER-01
- USER-02

Conversations

USER-01:HELLO, HOW ARE YOU?

USER-02:I AM FINE. WHAT ABOUT YOU?

hi