

PRAKTIK PER. 5

Langkah 1: Setup Project

```
npm create vite@latest praktik-5 -- --template react
```

Langkah 2: Komponen Event Handling Demo

File: src/components/EventDemo.jsx



EventDemo.jsx

```
1 import React, { useState } from 'react';
2
3 const EventDemo = () => {
4     const [message, setMessage] = useState('');
5     const [position, setPosition] = useState({ x: 0, y: 0 });
6     const [keyInfo, setKeyInfo] = useState('');
7
8     // Click Events
9     const handleClick = () => {
10         setMessage('Tombol diklik!');
11     };
12
13     const handleDoubleClick = () => {
14         setMessage('Double click terdeteksi!');
15     };
16
17     // Mouse Events
18     const handleMouseMove = (event) => {
19         setPosition({
20             x: event.clientX,
21             y: event.clientY
22         });
23     };
24
25     const handleMouseEnter = () => {
26         setMessage('Mouse masuk area!');
27     };
28
29     const handleMouseLeave = () => {
30         setMessage('Mouse keluar area!');
31     };
32
33     // Keyboard Events
34     const handleKeyDown = (event) => {
35         setKeyInfo(`Key: ${event.key}, Code: ${event.code}`);
36     };
37
38     const handleKeyUp = () => {
39         setTimeout(() => setKeyInfo(''), 1000);
40     };
41
42     // Context Menu
43     const handleContextMenu = (event) => {
44         event.preventDefault();
45         setMessage('Context menu dicegah!');
46     };
47
48     return (
49         <div className="event-demo">
```

```
50      <h2>Demo Penanganan Event</h2>
51
52      {/* Button Events */}
53      <div className="button-group">
54          <button onClick={handleClick} className="btn btn-primary">
55              Klik Saya
56          </button>
57
58          <button onDoubleClick={handleDoubleClick} className="btn btn-
secondary">
59              Double Klik Saya
60          </button>
61
62          <button onContextMenu={handleContextMenu} className="btn btn-
warning">
63              Right Click Saya
64          </button>
65      </div>
66
67      {/* Mouse Area */}
68      <div
69          className="mouse-area"
70          onMouseMove={handleMouseMove}
71          onMouseEnter={handleMouseEnter}
72          onMouseLeave={handleMouseLeave}
73      >
74          <h3>Area Interaksi Mouse</h3>
75          <p>Gerakkan mouse di area ini</p>
76          <p>Posisi: X: {position.x}, Y: {position.y}</p>
77      </div>
78
79      {/* Keyboard Input */}
80      <div className="keyboard-demo">
81          <h3>Demo Keyboard Events</h3>
82          <input
83              type="text"
84              placeholder="Ketik sesuatu di sini..."
85              onKeyDown={handleKeyDown}
86              onKeyUp={handleKeyUp}
87              className="keyboard-input"
88          />
89          {keyInfo && <p className="key-info">{keyInfo}</p>}
90      </div>
91
92      {/* Message Display */}
93      {message && (
94          <div className="message-box">
95              {message}
96          </div>
97      )}
98      </div>
99  );
100 };
101
```

```
102 export default EventDemo;
```

Langkah 3: Komponen Form Handling Demo

File: src/components/FormDemo.jsx



FormDemo.jsx

```
1 import React, { useState } from 'react';
2
3 const FormDemo = () => {
4     const [formData, setFormData] = useState({
5         nama: '',
6         email: '',
7         password: '',
8         usia: '',
9         jurusan: '',
10        semester: '1',
11        newsletter: false,
12        gender: ''
13    });
14
15    const [errors, setErrors] = useState({});
16    const [isSubmitted, setIsSubmitted] = useState(false);
17    const [touched, setTouched] = useState({});
18
19    // Handle Input Change
20    const handleChange = (event) => {
21        const { name, value, type, checked } = event.target;
22
23        setFormData(prevData => ({
24            ...prevData,
25            [name]: type === 'checkbox' ? checked : value
26        }));
27
28        // Clear error ketika user mulai mengetik
29        if (errors[name]) {
30            setErrors(prevErrors => ({
31                ...prevErrors,
32                [name]: ''
33            }));
34        }
35    };
36
37    // Handle Blur (ketika input kehilangan fokus)
38    const handleBlur = (event) => {
39        const { name } = event.target;
40        setTouched(prevTouched => ({
41            ...prevTouched,
42            [name]: true
43        }));
44    };
45
46    // Validasi Form
47    const validateForm = () => {
48        const newErrors = {};
49
```

```
50     // Validasi Nama
51     if (!formData.nama.trim()) {
52         newErrors.nama = 'Nama lengkap harus diisi';
53     } else if (formData.nama.trim().length < 2) {
54         newErrors.nama = 'Nama minimal 2 karakter';
55     }
56
57     // Validasi Email
58     if (!formData.email) {
59         newErrors.email = 'Email harus diisi';
60     } else if (!/\S+@\S+\.\S+/.test(formData.email)) {
61         newErrors.email = 'Format email tidak valid';
62     }
63
64     // Validasi Password
65     if (!formData.password) {
66         newErrors.password = 'Password harus diisi';
67     } else if (formData.password.length < 6) {
68         newErrors.password = 'Password minimal 6 karakter';
69     }
70
71     // Validasi Usia
72     if (!formData.usia) {
73         newErrors.usia = 'Usia harus diisi';
74     } else if (formData.usia < 17 || formData.usia > 60) {
75         newErrors.usia = 'Usia harus antara 17 - 60 tahun';
76     }
77
78     // Validasi Jurusan
79     if (!formData.jurusan) {
80         newErrors.jurusan = 'Jurusan harus dipilih';
81     }
82
83     // Validasi Gender
84     if (!formData.gender) {
85         newErrors.gender = 'Jenis kelamin harus dipilih';
86     }
87
88     setErrors(newErrors);
89     return Object.keys(newErrors).length === 0;
90 };
91
92 // Handle Form Submit
93 const handleSubmit = (event) => {
94     event.preventDefault();
95
96     if (validateForm()) {
97         console.log('Data form berhasil:', formData);
98         setIsSubmitted(true);
99         setTimeout(() => {
100             setIsSubmitted(false);
101         }, 3000);
102     }
103};
```

```
104
105 // Reset Form
106 const handleReset = () => {
107     setFormData({
108         nama: '',
109         email: '',
110         password: '',
111         usia: '',
112         jurusan: '',
113         semester: '1',
114         newsletter: false,
115         gender: ''
116     });
117     setErrors({});
118     setTouched({});
119 };
120
121 // Check jika field memiliki error dan sudah di-touch
122 const hasError = (fieldName) => {
123     return errors[fieldName] && touched[fieldName];
124 };
```



FormDemo.jsx

```
1      return (
2          <div className="form-demo">
3              <h2>Demo Form Handling</h2>
4
5              {isSubmitted && (
6                  <div className="success-message">
7                      Form berhasil dikirim! Data sudah tersimpan.
8                  </div>
9              )}
10
11
12          <form onSubmit={handleSubmit} onReset={handleReset}>
13              {/* Nama Lengkap */}
14              <div className="form-group">
15                  <label htmlFor="nama" className="form-label">
16                      Nama Lengkap *
17                  </label>
18                  <input
19                      type="text"
20                      id="nama"
21                      name="nama"
22                      value={formData.nama}
23                      onChange={handleChange}
24                      onBlur={handleBlur}
25                      className={`${'form-input ${hasError('nama') ? 'error' : ''}}`}>
26                      placeholder="Masukkan nama lengkap"
27                  />
28                  {hasError('nama') && <span className="error-text">
29                      {errors.nama}</span>}
30                  </div>
31
32                  {/* Email */}
33                  <div className="form-group">
34                      <label htmlFor="email" className="form-label">
35                          Email *
36                      </label>
37                      <input
38                          type="email"
39                          id="email"
40                          name="email"
41                          value={formData.email}
42                          onChange={handleChange}
43                          onBlur={handleBlur}
44                          className={`${'form-input ${hasError('email') ? 'error' : ''}}`}>
45                          placeholder="contoh@email.com"
46                  />
```

```
46          {hasError('email') && <span className="error-text">
47            {errors.email}</span>
48          </div>
49
50          {/* Password */}
51          <div className="form-group">
52            <label htmlFor="password" className="form-label">
53              Password *
54            </label>
55            <input
56              type="password"
57              id="password"
58              name="password"
59              value={formData.password}
60              onChange={handleChange}
61              onBlur={handleBlur}
62              className={`${'form-input ${hasError('password') ? 'error' : ''}}`}
63              placeholder="Minimal 6 karakter"
64            />
65            {hasError('password') && <span className="error-text">
66              {errors.password}</span>
67            </div>
68
69            {/* Usia */}
70            <div className="form-group">
71              <label htmlFor="usia" className="form-label">
72                Usia *
73              </label>
74              <input
75                type="number"
76                id="usia"
77                name="usia"
78                value={formData.usia}
79                onChange={handleChange}
80                onBlur={handleBlur}
81                className={`${'form-input ${hasError('usia') ? 'error' : ''}}`}
82                placeholder="17 - 60"
83                min="17"
84                max="60"
85              />
86              {hasError('usia') && <span className="error-text">
87                {errors.usia}</span>
88              </div>
```



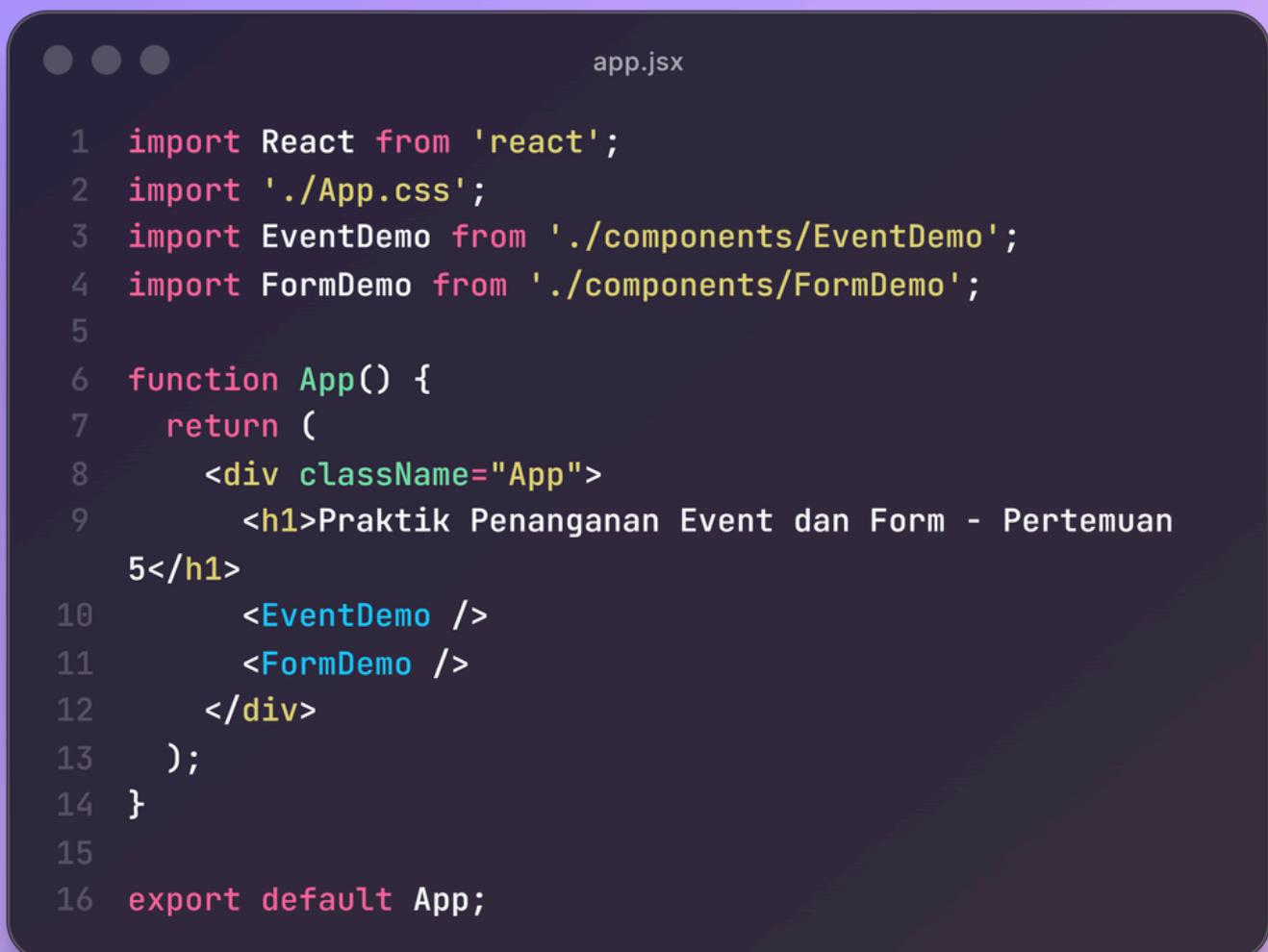
FormDemo.jsx

```
1      {/* Gender */}
2      <div className="form-group">
3          <label className="form-label">Jenis Kelamin </label>
4          <div className="radio-group">
5              <label className="radio-label">
6                  <input
7                      type="radio"
8                      name="gender"
9                      value="Laki-laki"
10                     checked={formData.gender === 'Laki-laki'}
11                     onChange={handleChange}
12                 />
13                 Laki-laki
14             </label>
15             <label className="radio-label">
16                 <input
17                     type="radio"
18                     name="gender"
19                     value="Perempuan"
20                     checked={formData.gender === 'Perempuan'}
21                     onChange={handleChange}
22                 />
23                 Perempuan
24             </label>
25         </div>
26         {hasError('gender') && <span className="error-text">
27             {errors.gender}</span>}
28         </div>
29
30         {/* Jurusan */}
31         <div className="form-group">
32             <label htmlFor="jurusan" className="form-label">
33                 Jurusan *
34             </label>
35             <select
36                 id="jurusan"
37                 name="jurusan"
38                 value={formData.jurusan}
39                 onChange={handleChange}
40                 onBlur={handleBlur}
41                 className={`form-select ${hasError('jurusan') ? 'error' : ''}}>
42                 >
43                     <option value="">Pilih Jurusan</option>
44                     <option value="Teknik Informatika">Teknik
Informatika</option>
45                     <option value="Sistem Informasi">Sistem
Informasi</option>
```

```
46          <option value="Teknik Komputer">Teknik  
47          Komputer</option>  
48      </select>  
49      {hasError('jurusan') && <span className="error-text">  
50      {errors.jurusan}</span>}  
51  </div>  
52  {/* Semester */}  
53  <div className="form-group">  
54      <label htmlFor="semester" className="form-label">  
55          Semester  
56      </label>  
57      <select  
58          id="semester"  
59          name="semester"  
60          value={formData.semester}  
61          onChange={handleChange}  
62          className="form-select"  
63      >  
64          {[1, 2, 3, 4, 5, 6, 7, 8].map(sem => (  
65              <option key={sem} value={sem.toString()}>  
66                  Semester {sem}  
67              </option>  
68          ))}  
69      </select>  
70  </div>  
71  {/* Newsletter */}  
72  <div className="form-group checkbox-group">  
73      <label className="checkbox-label">  
74          <input  
75              type="checkbox"  
76              name="newsletter"  
77              checked={formData.newsletter}  
78              onChange={handleChange}  
79          />  
80          Saya ingin berlangganan newsletter  
81      </label>  
82  </div>  
83  
84  {/* Form Buttons */}  
85  <div className="form-buttons">  
86      <button type="submit" className="btn btn-success">  
87          Daftar  
88      </button>  
89      <button type="reset" className="btn btn-secondary">  
90          Reset  
91      </button>  
92  </div>  
93  </form>  
94  
95  {/* Form Data Preview */}  
96  <div className="data-preview">
```

```
98             <h3>Preview Data Form:</h3>
99             <pre>{JSON.stringify(formData, null, 2)}</pre>
100         </div>
101     </div>
102   );
103 };
104
105 export default FormDemo;
```

Langkah 4: App.jsx



The screenshot shows a code editor window with a dark theme. The title bar says "app.jsx". The code in the editor is:

```
1 import React from 'react';
2 import './App.css';
3 import EventDemo from './components/EventDemo';
4 import FormDemo from './components/FormDemo';
5
6 function App() {
7   return (
8     <div className="App">
9       <h1>Praktik Penanganan Event dan Form - Pertemuan
10      5</h1>
11      <EventDemo />
12      <FormDemo />
13    </div>
14  );
15
16 export default App;
```

App.css



App.css

```
1 .App {  
2   max-width: 1200px;  
3   margin: 0 auto;  
4   padding: 20px;  
5   font-family: 'Segoe UI', Tahoma, Geneva, Verdana, sans-serif;  
6 }  
7  
8 /* Event Demo Styles */  
9 .event-demo {  
10   background: #f8f9fa;  
11   padding: 20px;  
12   border-radius: 10px;  
13   margin-bottom: 30px;  
14   border-left: 4px solid #007bff;  
15 }  
16  
17 .button-group {  
18   display: flex;  
19   gap: 10px;  
20   margin: 20px 0;  
21   flex-wrap: wrap;  
22 }  
23  
24 .btn {  
25   padding: 10px 20px;  
26   border: none;  
27   border-radius: 5px;  
28   cursor: pointer;  
29   font-weight: bold;  
30   transition: all 0.3s ease;  
31 }  
32  
33 .btn-primary {  
34   background: #007bff;  
35   color: white;  
36 }  
37  
38 .btn-secondary {  
39   background: #6c757d;  
40   color: white;  
41 }  
42  
43 .btn-link {  
44   color: #007bff;  
45   text-decoration: underline;  
46   background-color: transparent;  
47   border-color: transparent;
```

```
43 .btn-warning {
44   background: #ffc107;
45   color: black;
46 }
47
48 .btn-success {
49   background: #28a745;
50   color: white;
51 }
52
53 .btn:hover {
54   opacity: 0.9;
55   transform: translateY(-2px);
56 }
57
58 .mouse-area {
59   background: #e9ecef;
60   padding: 20px;
61   border-radius: 8px;
62   margin: 20px 0;
63   border: 2px dashed #dee2e6;
64   text-align: center;
65 }
66
67 .keyboard-demo {
68   margin: 20px 0;
69 }
70
71 .keyboard-input {
72   width: 100%;
73   max-width: 400px;
74   padding: 10px;
75   border: 2px solid #dee2e6;
76   border-radius: 5px;
77   font-size: 16px;
78 }
79
80 .key-info {
81   background: #d1ecf1;
82   padding: 10px;
83   border-radius: 5px;
84   margin-top: 10px;
85 }
86
87 .message-box {
88   background: #d4edda;
89   color: #155724;
90   padding: 15px;
```

```
    padding: 10px;
91    border-radius: 5px;
92    margin: 15px 0;
93    border: 1px solid #c3e6cb;
94  }
95
96 /* Form Demo Styles */
97 .form-demo {
98   background: #fff;
99   padding: 20px;
100  border-radius: 10px;
101  border: 1px solid #dee2e6;
102  box-shadow: 0 2px 10px rgba(0, 0, 0, 0.1);
103 }
104
105 .form-container {
106   display: grid;
107   gap: 20px;
108   margin-bottom: 30px;
109 }
110
111 .form-group {
112   display: flex;
113   flex-direction: column;
114 }
115
116 .form-label {
117   font-weight: bold;
118   margin-bottom: 5px;
119   color: #495057;
120 }
121
122 .form-input,
123 .form-select {
124   padding: 12px;
125   border: 2px solid #e9ecf;
126   border-radius: 5px;
127   font-size: 16px;
128   transition: border-color 0.3s ease;
129 }
130
131 .form-input:focus,
132 .form-select:focus {
133   outline: none;
134   border-color: #007bff;
135 }
136
137 .form-input.error,
```

```
138 .form-select.error {  
139   border-color: #dc3545;  
140 }  
141  
142 .error-text {  
143   color: #dc3545;  
144   font-size: 14px;  
145   margin-top: 5px;  
146 }  
147  
148 .radio-group {  
149   display: flex;  
150   gap: 20px;  
151   margin-top: 5px;  
152 }  
153  
154 .radio-label,  
155 .checkbox-label {  
156   display: flex;  
157   align-items: center;  
158   gap: 8px;  
159   cursor: pointer;  
160 }  
161  
162 .checkbox-group {  
163   margin-top: 10px;  
164 }  
165  
166 .form-buttons {  
167   display: flex;  
168   gap: 10px;  
169   margin-top: 20px;  
170 }  
171  
172 .success-message {  
173   background: #d4edda;  
174   color: #155724;  
175   padding: 15px;  
176   border-radius: 5px;  
177   margin-bottom: 20px;  
178   border: 1px solid #c3e6cb;  
179   text-align: center;  
180   font-weight: bold;  
181 }  
182  
183 .data-preview {  
184   background: #f8f9fa;  
185   padding: 15px;  
186 }
```

```
185   padding: 15px;
186   border-radius: 5px;
187   margin-top: 20px;
188 }
189
190 .data-preview pre {
191   background: #e9ecf;
192   padding: 15px;
193   border-radius: 5px;
194   overflow-x: auto;
195   font-size: 14px;
196 }
197
198 /* Responsive Design */
199 @media (max-width: 768px) {
200   .button-group {
201     flex-direction: column;
202   }
203
204   .form-buttons {
205     flex-direction: column;
206   }
207
208   .radio-group {
209     flex-direction: column;
210     gap: 10px;
211   }
212 }
```

Hasil

Praktik Penanganan Event dan Form - Pertemuan 5

Demo Penanganan Event

Klik Saya

Double Klik Saya

Right Click Saya

Area Interaksi Mouse

Gerakkan mouse di area ini

Posisi: X: 432, Y: 433

Demo Keyboard Events

Ketik sesuatu di sini...

Mouse masuk area!

Demo Form Handling

Nama Lengkap *

Insan

Email *

insan@gmail.com

Password *

.....

Usia *

17

Jenis Kelamin *

Laki-laki Perempuan

Jurusan *

Teknik Informatika