

Ex No: 5

REACT CLASS COMPONENTS

Aim:

To execute problems on React Class Components.

1. Emergency Alert System

Scenario: Build a city-wide emergency alert system where alerts are dynamically displayed based on severity.

Tasks:

1. Create an AlertCard class component that:
 - Accepts title, message, and severity as props. Severity can be "low", "moderate", or "high".
 - Dynamically assigns a background color (e.g., green for "low", orange for "moderate", red for "high").
2. Add a "Snooze Alert" button that hides the alert for 10 seconds and then displays it again (use setTimeout and state).
3. If the message exceeds 150 characters, display a "Show More" button that toggles the full text when clicked.

Code:

Emergency Alert.js

```
import React, { Component } from "react";

class AlertCard extends Component {
  constructor(props) {
    super(props);
    this.state = {
      visible: true,
      expanded: false,
    };
  }

  getBackgroundColor() {
    const { severity } = this.props;
    switch (severity) {
      case "low":
        return "green";
```

```

    case "moderate":
        return "orange";
    case "high":
        return "red";
    default:
        return "gray";
    }
}

snoozeAlert = () => {
    this.setState({ visible: false });
    setTimeout(() => {
        this.setState({ visible: true });
    }, 10000);
};

toggleExpand = () => {
    this.setState((prevState) => ({ expanded: !prevState.expanded }));
};

render() {
    const { title, message = "" } = this.props; // Default message to an empty string
    const { visible, expanded } = this.state;
    const bgColor = this.getBackgroundColor();
    if (!visible) return null;
    return (
        <div
            style={{
                backgroundColor: bgColor,
                color: "white",
                padding: "15px",
                margin: "10px 0",
                borderRadius: "5px",
            }}
        >

```

```

<h3>{title}</h3>

<p>
  {expanded || message.length <= 150
    ? message
    : message.substring(0, 150) + "..."}
</p>

{message.length > 150 && (
  <button onClick={this.toggleExpand} style={{ marginRight: "10px" }}>
    {expanded ? "Show Less" : "Show More"}
  </button>
)}

<button onClick={this.snoozeAlert}>Snooze Alert</button>
</div>

);
}
}

export default AlertCard;

```

App.js

```

import './App.css';
import AlertCard from './Components/Emergency Alert';

function App() {
  return (
    <div className="App">
      <AlertCard
        title="Severe Weather Warning"
        message="A severe storm is approaching. Please stay indoors and avoid unnecessary travel.
Ensure you have emergency supplies."
        severity="high"
      />
      <AlertCard
        title="Traffic Alert"
        message="Heavy traffic congestion due to an accident on Main Street. Expect delays."

```

```

        severity="moderate"
      />

      <AlertCard

        title="Low Battery Alert"

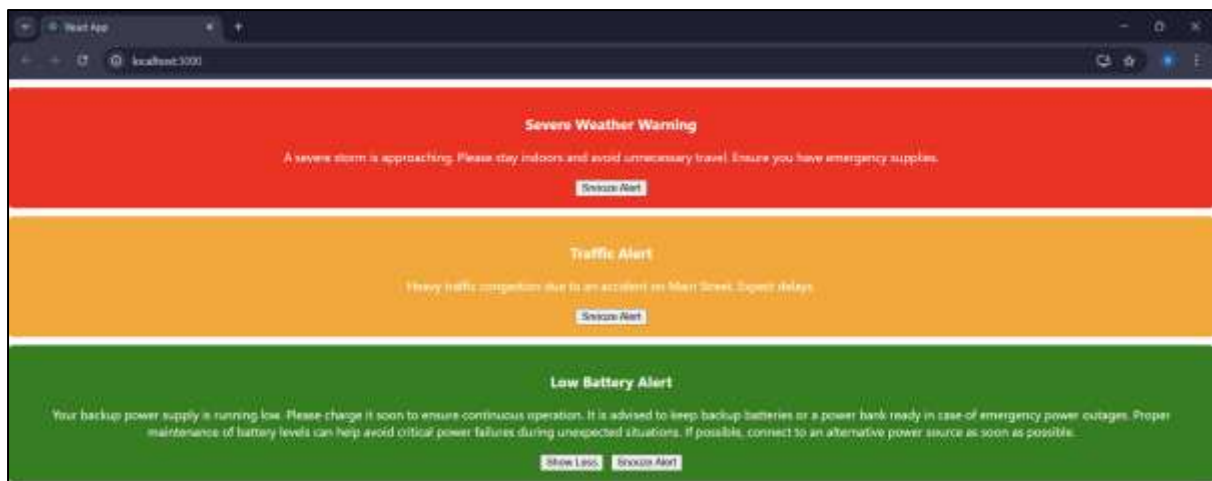
        message="Your backup power supply is running low. Please charge it soon to ensure continuous
operation. It is advised to keep backup batteries or a power bank ready in case of emergency power
outages. Proper maintenance of battery levels can help avoid critical power failures during unexpected
situations. If possible, connect to an alternative power source as soon as possible."

        severity="low"
      />
    </div>
  );
}

export default App;

```

Output:



2. Pet Adoption Platform

Scenario: Build a platform for showcasing pets available for adoption.

Tasks:

- Create a PetCard class component that:
 - Displays the pet's name, age, species, and a brief description as props.
 - Adds a badge for "Young" pets if the age is less than 2 years.
- Add a "Toggle Details" button that shows or hides the full description of the pet.
- If the species is "dog", display a 🐶 icon; if "cat", display a 🐱 icon. For other species, show a generic 🐾 icon.

Code:**Pet Adoption.js**

```
import React, { Component } from "react";

class PetCard extends Component {
  constructor(props) {
    super(props);
    this.state = {
      showDetails: false,
    };
  }

  toggleDetails = () => {
    this.setState((prevState) => ({ showDetails: !prevState.showDetails }));
  };

  getSpeciesIcon() {
    const { species } = this.props;
    if (species.toLowerCase() === "dog") return "🐶";
    if (species.toLowerCase() === "cat") return "🐱";
    return "🐾";
  }

  render() {
    const { name, age, species, description } = this.props;
    const { showDetails } = this.state;
    return (
      <div
        style={{
          border: "1px solid #ddd",
          borderRadius: "8px",
          padding: "15px",
          margin: "10px",
          width: "250px",
          textAlign: "center",
        }}
      >
```

```

    >
    <h2>
      {this.getSpeciesIcon()} {name}
    </h2>
    <p>Species: {species}</p>
    <p>Age: {age} {age < 2 && <span style={{ color: "green", fontWeight: "bold" }}>
(Young)</span></p>
    {showDetails ? <p>{description}</p> : <p>{description.substring(0, 100)}...</p>}
    <button onClick={this.toggleDetails} style={{ marginTop: "10px" }}>
      {showDetails ? "Hide Details" : "Show Details"}
    </button>
  </div>
);
}
}
export default PetCard;

```

App.js

```

import './App.css';
import PetCard from './Components/Pet Adoption';
function App() {
  return (
    <div style={{ display: "flex", gap: "20px", flexWrap: "wrap", alignItems: "center", justifyContent:
"center" }}>
      <PetCard
        name="Buddy"
        age={1}
        species="dog"
        description="Buddy is a playful golden retriever who loves to run in the park and play fetch. He
is friendly and great with kids."
      />
      <PetCard
        name="Whiskers"
        age={3}

```

species="cat"

description="Whiskers is a calm and affectionate cat who enjoys lounging by the window and getting head scratches."

/>

<PetCard

name="Charlie"

age={4}

species="rabbit"

description="Charlie is an energetic rabbit who loves hopping around and nibbling on fresh vegetables. A great companion for pet lovers!"

/>

<PetCard

name="Luna"

age={0.5}

species="dog"

description="Luna is a friendly puppy who loves to play with toys and snuggle. She is looking for a loving home!"

/>

<PetCard

name="Mittens"

age={2}

species="cat"

description="Mittens is an adorable and independent cat. She enjoys watching birds from the window and taking long naps."

/>

<PetCard

name="Bella"

age={1}

species="parrot"

description="Bella is a talkative parrot who loves to mimic sounds and interact with people. She is colorful and full of energy!"

/>

<PetCard

name="Shadow"

```

    age={5}

    species="dog"

    description="Shadow is a loyal and protective German Shepherd. He enjoys long walks and
    playing fetch at the park."

  />

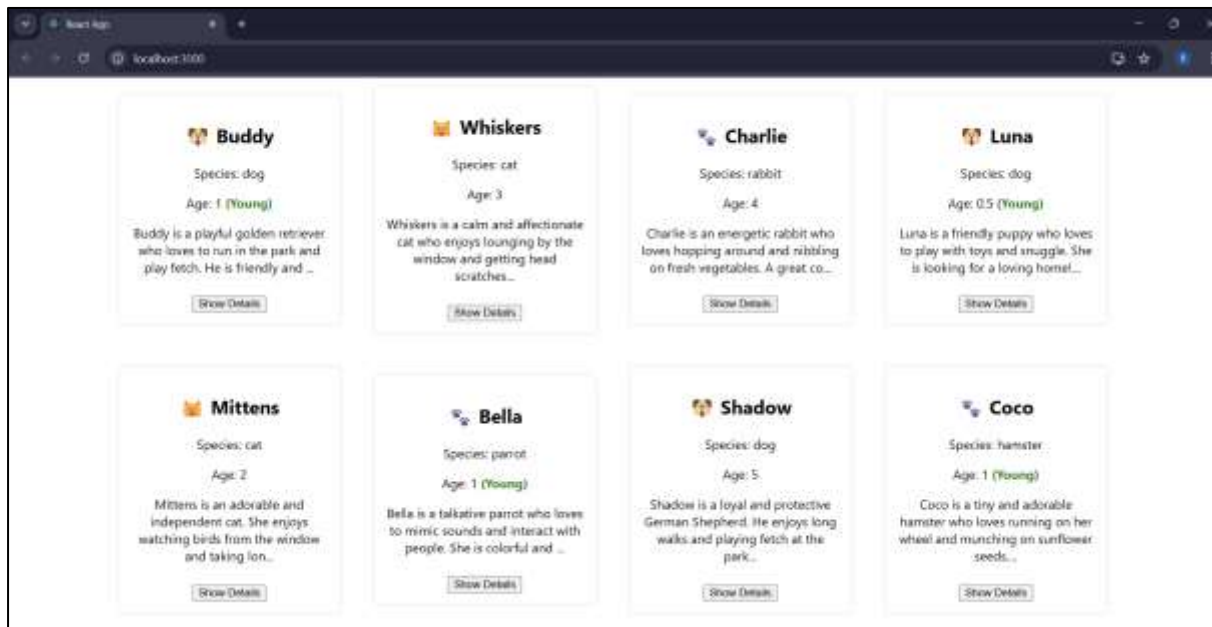
</div>

);
}

export default App;

```

Output:



Problem Identification (5)	Execution (5)	Time Management (5)	Viva (5)	Total (20)

Result:

Thus the problems on React Class Components were executed successfully and the output was verified.