

# **Wprowadzenie do React.js**

Biblioteka JavaScript do budowania nowoczesnych interfejsów.



# Czym jest React

- React to biblioteka JavaScript do tworzenia interfejsów
- Stworzona przez Facebooka w 2013 roku
- Popularna wśród firm i programistów na całym świecie



# Czym charakteryzuje się React



**Deklaratywność** - łatwiejsze zarządzania UI

**Jednokierunkowy przepływ danych** - łatwiejsza kontrola nad stanem aplikacji

**Hooki** - nowoczesny sposób zarządzania stanem i efektami

**Komponentowość** - aplikacje składają się z małych, niezależnych komponentów

**Wirtualny DOM** - szybsze renderowanie zmian w interfejsie

\*fajna tematyczna grafika\*

# Dlaczego warto używać Reacta?



Komponentowy model ułatwia zarządzanie kodem

---

Wysoka wydajność dzięki Wirtualnemu DOM

---

Szerokie wsparcie społeczności i rozwój ekosystemu

# JSX – język szablonów w React

JSX (JavaScript XML) – składnia łącząca HTML i JavaScript  
Ułatwia tworzenie dynamicznych interfejsów

```
const element = <h1>Hello, React!</h1>;
```

# Komponenty – podstawowa jednostka React

- React składa się z komponentów (funkcjonalnych i klasowych)
- Komponenty można wielokrotnie używać w różnych miejscach

```
function Hello() {  
  return <h1>Witaj w React!</h1>;  
}
```

# Props – przekazywanie danych

- Props (właściwości) pozwalają przekazywać dane między komponentami
- Są one tylko do odczytu

```
function Welcome(props) {  
  return <h1>Witaj, {props.name}!</h1>;  
}
```

# Stan (State) – dynamiczne dane w React

- Stan pozwala przechowywać i aktualizować dynamiczne dane
- React aktualizuje UI po każdej zmianie stanu

```
function Counter() {  
  const [count, setCount] = React.useState(0);  
  return (  
    <div>  
      <p>Licznik: {count}</p>  
      <button onClick={()=>setCount(count+1)}>Zwiększ</button>  
    </div>);  
}
```

# Wirtualny DOM – jak React optymalizuje wydajność?

- Wirtualny DOM to kopia rzeczywistego DOM
- React aktualizuje tylko zmienione elementy, nie cały dokument



