

# VUE COMPONENTS

---

## TEORIA

1. Komponenty stanowią kluczowy składnik vue.js, umożliwiając tworzenie jednostek kodu, które mogą być wielokrotnie wykorzystywane zarówno w tworzonej, jak i przyszłych aplikacjach. Każdy stworzony komponent posiada unikalną nazwę. Zwróć uwagę na prawidłowe nazewnictwo tworzonych komponentów, podane w poniższych materiałach.
2. Zapoznaj się z materiałem filmowym prezentującym podstawy tworzenia komponentów:
3. Zapoznaj się z materiałem filmowym prezentującym podstawy tworzenia komponentów:

<https://vueschool.io/courses/vuejs-components-fundamentals?friend=vuejs>

<https://youtu.be/EAONyIKrZN0>

Materiał ten wchodzi w skład dostępnego kursu Vue.js:

<https://www.youtube.com/playlist?list=PL4cUxeGkcC9gQcYgihBoeQH7wiAyZNRya>

4. Zapoznaj się z materiałem dotyczącym podstaw tworzenia komponentów:

<https://vuejs.org/v2/guide/components.html>

5. Utrwal swoją wiedzę na temat tworzenia komponentów,

[https://www.tutorialspoint.com/vuejs/vuejs\\_components.htm](https://www.tutorialspoint.com/vuejs/vuejs_components.htm)

## PRAKTYKA

6. Aplikacja university.html wyświetla nazwę i adres uniwersytetu. Zmodyfikuj aplikację, aby dane były wyświetlane przy użyciu komponentu.
7. Utwórz aplikację films.html, w której wyświetl informacje o trzech ulubionych filmach. Każdy film opisz, podając: tytuł filmu, reżysera, rok produkcji. Wyświetl informację o każdym filmie używając odrębnego komponentu.
8. Odszukaj w Internecie, jakie elementy wchodzi w skład architektury komputera wg von Neumanna. Następnie utwórz aplikację architektura.html, a w niej komponent, który wyświetli listę składowych architektury von Neumanna. Nazwy składowych zapisz w komponencie (data). Do wyświetlenia nazw zastosuj dyrektywę v-for. Nad nazwami umieść tytuł listy (Architektura von Neumanna).

9. Utwórz aplikację binary.html, a w niej komponent, który wyświetla liczby naturalne w postaci binarnej. Zadaniem komponentu jest konwersja liczby dziesiętnej przekazanej do komponentu do wartości binarnej, a następnie wyświetlenie rezultatu w formacie, jak poniżej:

$$17(10) = 10001(2)$$

Korzystając z utworzonego komponentu, wyświetl wartości binarne dla liczb w systemie dziesiętnym: 8, 170, 204, 3855 (skorzystaj z utworzonego komponentu kilkakrotnie).

10. Utwórz aplikację charts.html, a w niej komponent, który przedstawi na wykresie dowolny zbiór wartości liczbowych. Dane liczbowe do wyświetlenia prześlij do komponentu. Dla wyświetlenia wykresu wykorzystaj dowolną bibliotekę do tworzenia wykresów, np. chart.js (wybierz dowolny typ wykresu, np. liniowy, kołowy czy kolumnowy):

<https://www.chartjs.org/>

11. Utwórz aplikację counter.html, a w niej komponent, który wyświetla stan licznika (liczba całkowita) oraz dwa przyciski: Dodaj oraz Odejmij. Kliknięcie przycisku Dodaj spowoduje zwiększenie stanu licznika o 1, a kliknięcie przycisku Odejmij spowoduje zmniejszenie stanu licznika o 1.

12. Utwórz aplikację api.html, a w niej komponent, który wyświetli dane z dowolnego serwisu udostępnianego w Internecie poprzez API. Możesz wykorzystać dowolne API, np. jedno z dostępnych poniżej lub dowolne inne:

<https://github.com/public-apis/public-apis>