

The background of the slide features a dark blue field with various tech-related icons. On the left, there's a stylized orange and blue figure of a person wearing a headset. In the center, there's a large white 'B' inside a purple rounded square. Below that, a blue square contains the white letters 'TS'. At the bottom right, there's an orange shield-shaped icon with a white 'S' and the word 'HTML' above it. The overall aesthetic is modern and tech-oriented.

Vanilla  
**JS**

# Frontend -Vanilla JS/TypeScript i HTML5/Bootstrap

Filip Kubala, Jakub Wiatr



Czym w ogóle  
jest ten cały  
**„frontend”**?





**Frontend** – rozwijanie graficznego interfejsu użytkownika strony internetowej, poprzez użycie **HTML**, **CSS** i **JavaScript**, dzięki czemu użytkownicy mogą oglądać i wchodzić w interakcję ze stroną.

Źródło: Wikipedia

# HTML



## Podstawy HTML5:

### Podstawy HTML

*HTML (HyperText Markup Language)* to język używany do tworzenia stron internetowych. Oto kilka podstawowych elementów:

1. **<!DOCTYPE html>**: Deklaracja typu dokumentu HTML.
2. **<html>**: Element korzenia dokumentu HTML.
3. **<head>**: Kontener dla metadanych, takich jak tytuł strony, linki do arkuszy stylów itp.
4. **<title>**: Tytuł strony, wyświetlany w pasku przeglądarki.
5. **<body>**: Zawartość strony, takie jak tekst, obrazy, linki, itp.

Źródło: <https://www.browserstack.com/guide/top-html5-features>

# HTML



## Główne cechy HTML5:

- **Semantyka:** Zwięzłe i precyzyjne znaczniki dla lepszego zrozumienia struktury dokumentu
- **Audio i video:** Wbudowane wsparcie dla elementów audio i video, eliminując konieczność stosowania zewnętrznych wtyczek
- **Nowe znaczniki:** Wprowadzenie nowych, bardziej semantycznych znaczników, takich jak `<article>`, `<section>`, `<nav>` itp.
- **WebSockets:** Technologia umożliwiająca dwustronną komunikację pomiędzy klientem a serwerem w czasie rzeczywistym
- **Element Canvas:** Możliwość renderowania grafiki bezpośrednio na stronie

Źródło: <https://www.browserstack.com/guide/top-html5-features>

# HTML



## Podstawy HTML5:

W tym przykładzie:

- Strona ma tytuł "Moja Strona".
- Zawiera nagłówek pierwszego poziomu (h1).
- Posiada jeden paragraf tekstu.
- Zawiera link do przykładowej strony.
- Wyświetla obraz z pliku "obraz.jpg" z opisem "Opis obrazu".

```
<!DOCTYPE html>
<html>

<head>
  <title>Moja Strona</title>
</head>

<body>
  <h1>Witaj na mojej stronie!</h1>
  <p>To jest pierwszy paragraf mojej strony.</p>
  <a href="https://www.przykladowastrona.com">Przykładowy Link</a>
  
</body>

</html>
```

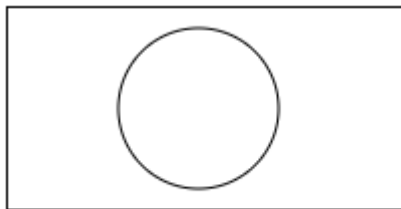
Źródło: <https://www.browserstack.com/guide/top-html5-features>

# HTML



## Przykłady zastosowania Canvas:

```
1  <!DOCTYPE html>
2  <html lang="en">
3  <head>
4  <title>HTML5</title>
5  </head>
6  <body>
7  <canvas id="Canvas" width="200" height="100" style="border:1px solid black;">Canvas</canvas>
8  </body>
9  <script>
10 var c = document.getElementById("Canvas");
11 var ctx = c.getContext("2d");
12 ctx.beginPath();
13 ctx.arc(95,50,40,0,2*Math.PI);
14 ctx.stroke();
15 </script>
16 </html>
```

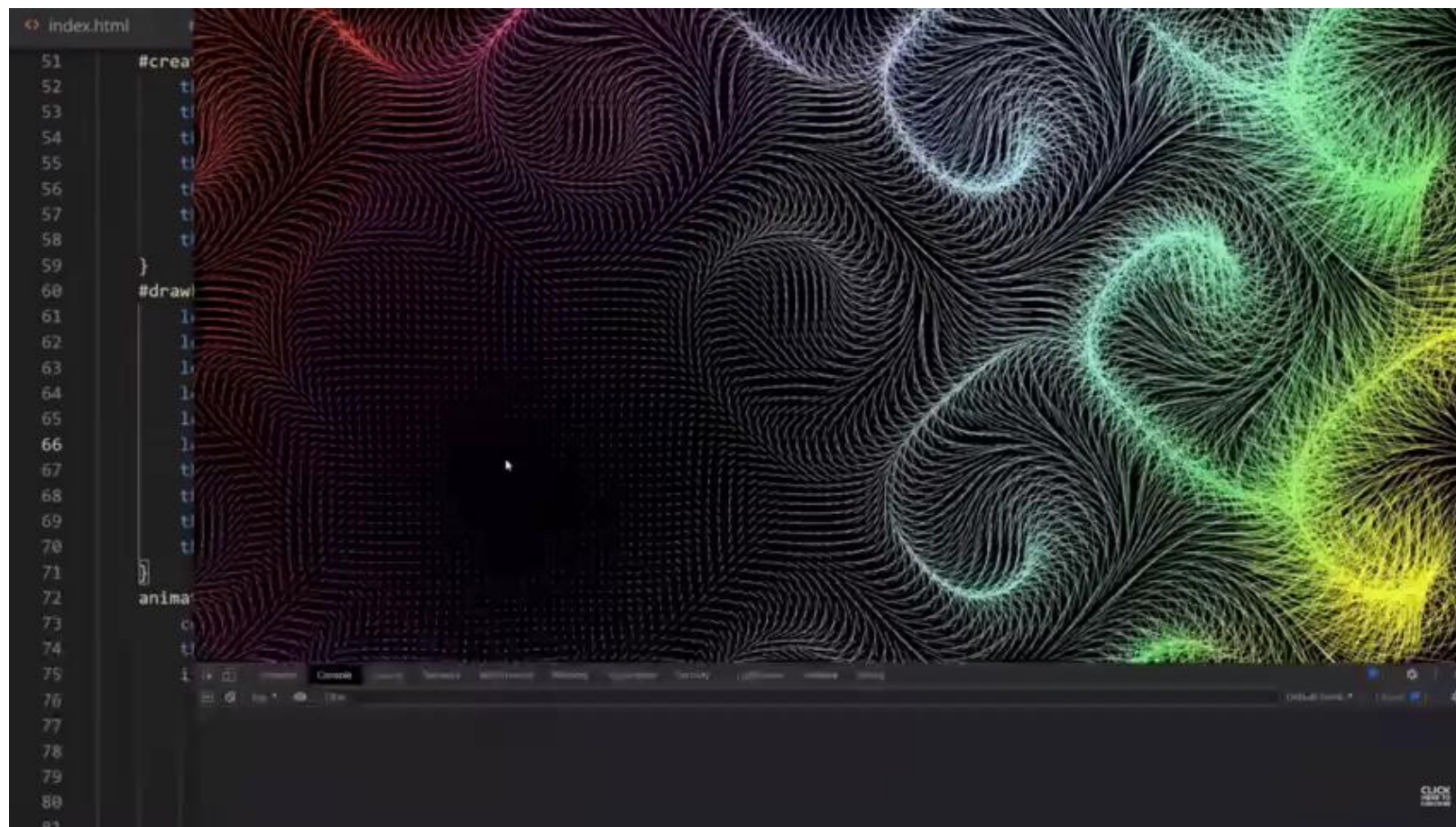




# HTML




## Przykłady zastosowania Canvas:



Źródło: <https://www.youtube.com/watch?v=uCH1ta5OUHw>



A stylized illustration on the left side of the slide. It shows a person with dark hair, wearing a blue shirt, sitting at a desk and working on a laptop. The person is wearing headphones. A large, purple Bootstrap logo, which is a square with a white 'B' inside, is overlaid on the person's head and upper body. The background of the illustration is dark blue with some faint, glowing lines and shapes, suggesting a digital or tech environment.

# Bootstrap – Co to jest ?

**Bootstrap** - to lubiany przez programistów, najpopularniejszy na świecie zbiór bibliotek open-source umożliwiający szybkie projektowanie oraz tworzenie kodu frontendu responsywnych stron internetowych. Założeniem technologii jest mobile first design - czyli skupienie się najpierw na wersji mobilnej. Bootstrap jest możliwy do zaadoptowania w niemal każdym projekcie strony internetowych.

<https://ks.pl/slownik/co-to-jest-bootstrap>

A stylized illustration on the left side of the slide. It shows a person with dark hair, wearing a blue shirt, sitting at a desk and working on a laptop. The person is rendered in a simplified, almost wireframe-like style with some color. Behind them is a large, semi-transparent Bootstrap logo, which is a purple square with a white 'B' inside. The background of the illustration is dark blue with some faint, glowing lines and shapes, suggesting a digital or tech environment.

# Czy warto używać Bootstrap?

Bootstrap jest podstawową biblioteką, po którą sięgają Front-End Developerzy, jeśli więc zamierzasz się w tym kierunku rozwijać, nie możesz tego frameworku ignorować. Bootstrap przyspiesza prace nad rozbudowanymi i złożonymi projektami, pozwalając na oszczędności w czasie realizacji - a to z kolei pozwala obniżyć również koszty i stać się konkurencyjnym na rynku. Choć nie wyręczy Cię całkowicie w pracy nad Twoim projektem, będzie bardzo dużym wsparciem, zwłaszcza wówczas, gdy nie masz jeszcze biegłości w kodowaniu.

<https://blog.strefakursow.pl/czym-jest-bootstrap-i-czy-warto-go-uzywac/>

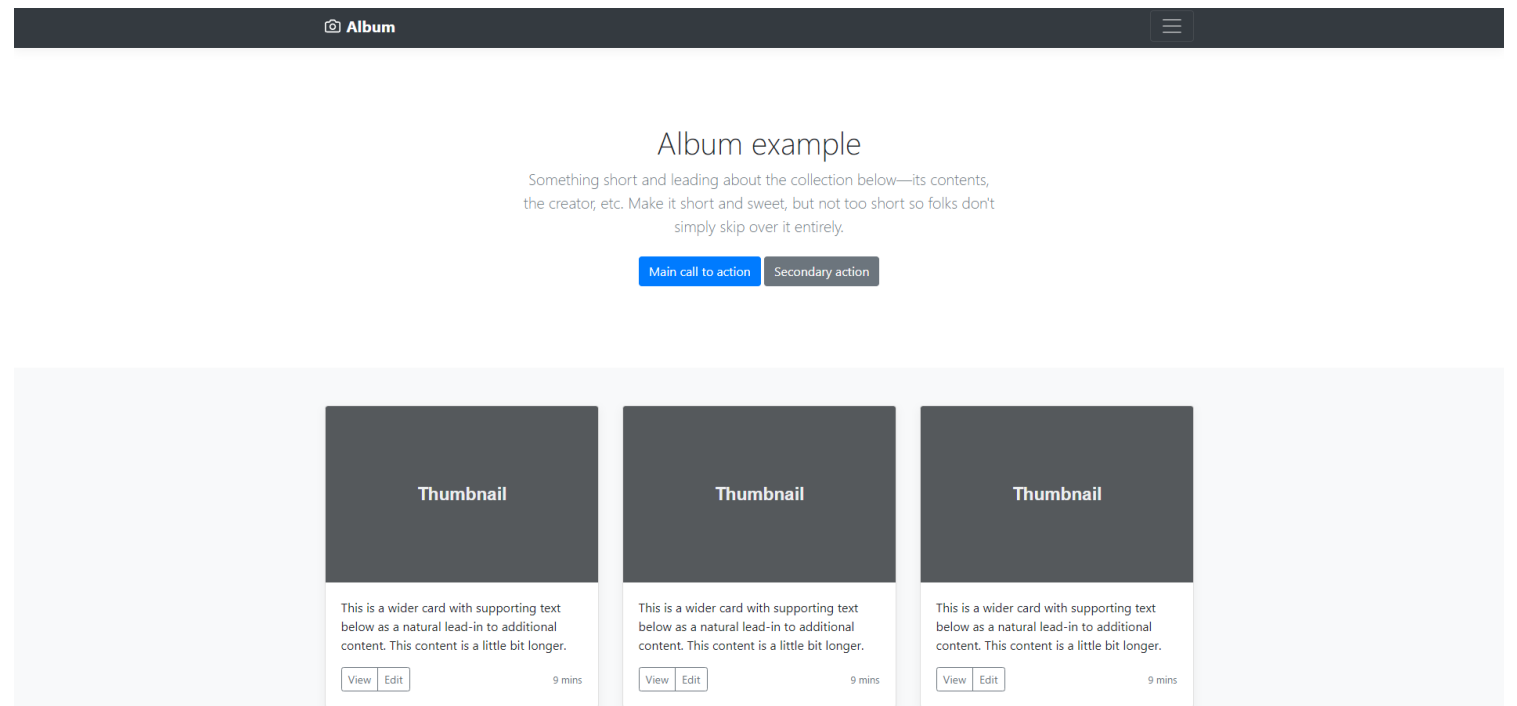


# Zalety Bootstrapa

Najważniejsze zalety JavaScript:

- 1.Szybki start:** Bootstrap dostarcza gotowe szablony i komponenty, co pozwala na szybkie rozpoczęcie projektu bez konieczności tworzenia wszystkiego od zera.
- 2.Responsywność:** Używając Bootstrapa możesz łatwo tworzyć responsywne strony, które dobrze wyglądają na ekranach o różnych rozdzielczościach
- 3.Łatwość w użyciu:** Bootstrap jest bardzo prosty w użyciu, każdy z podstawową wiedzą HTML, CSS i JavaScript może zacząć korzystać z Bootstrapa
- 4.Zgodność z przeglądarkami:** Bootstrap jest dostosowany pod nowoczesne przeglądarki i jest kompatybilny z większością z nich.
- 5.Open Source:** Bootstrap jest darmowy, każdy może z niego korzystać nic nie płacąc.

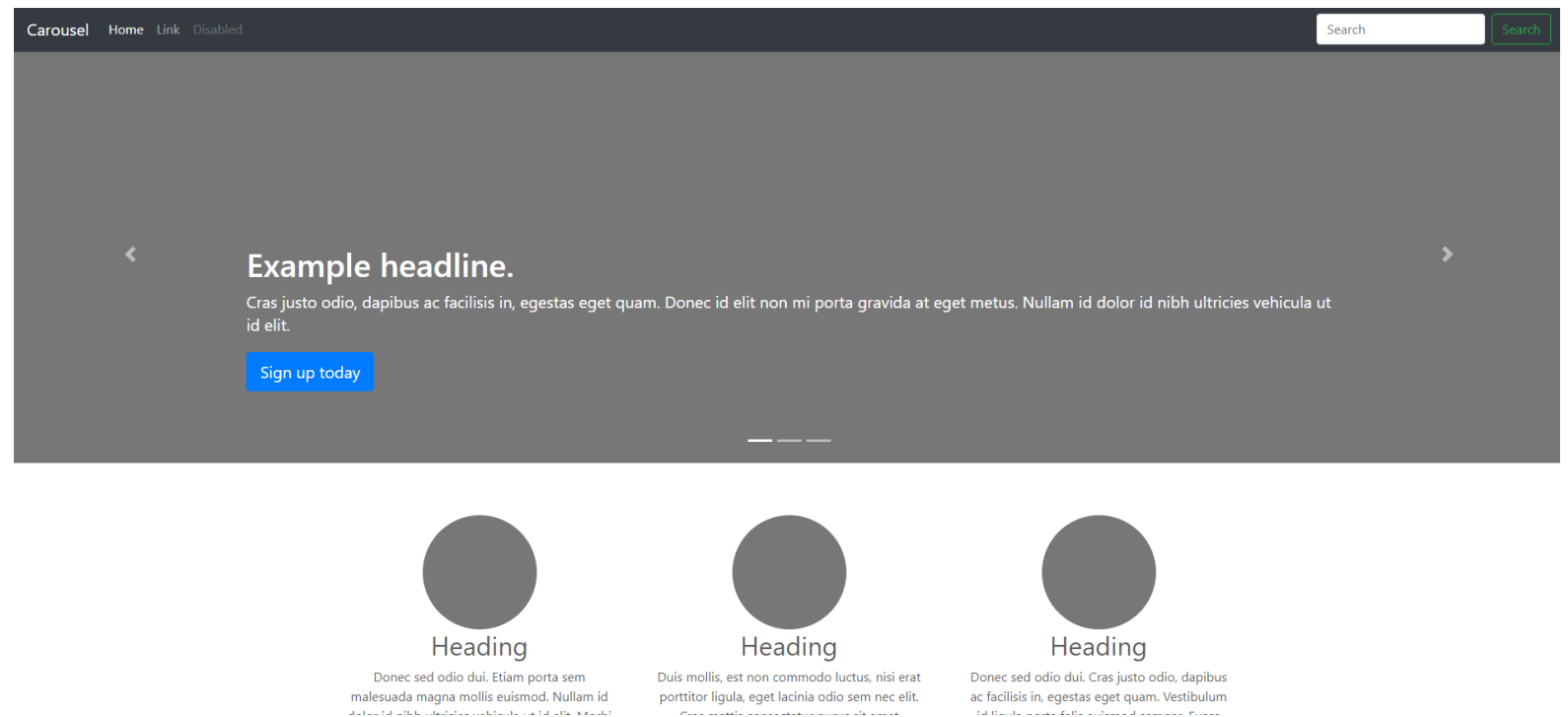
# Przykłady zastosowań





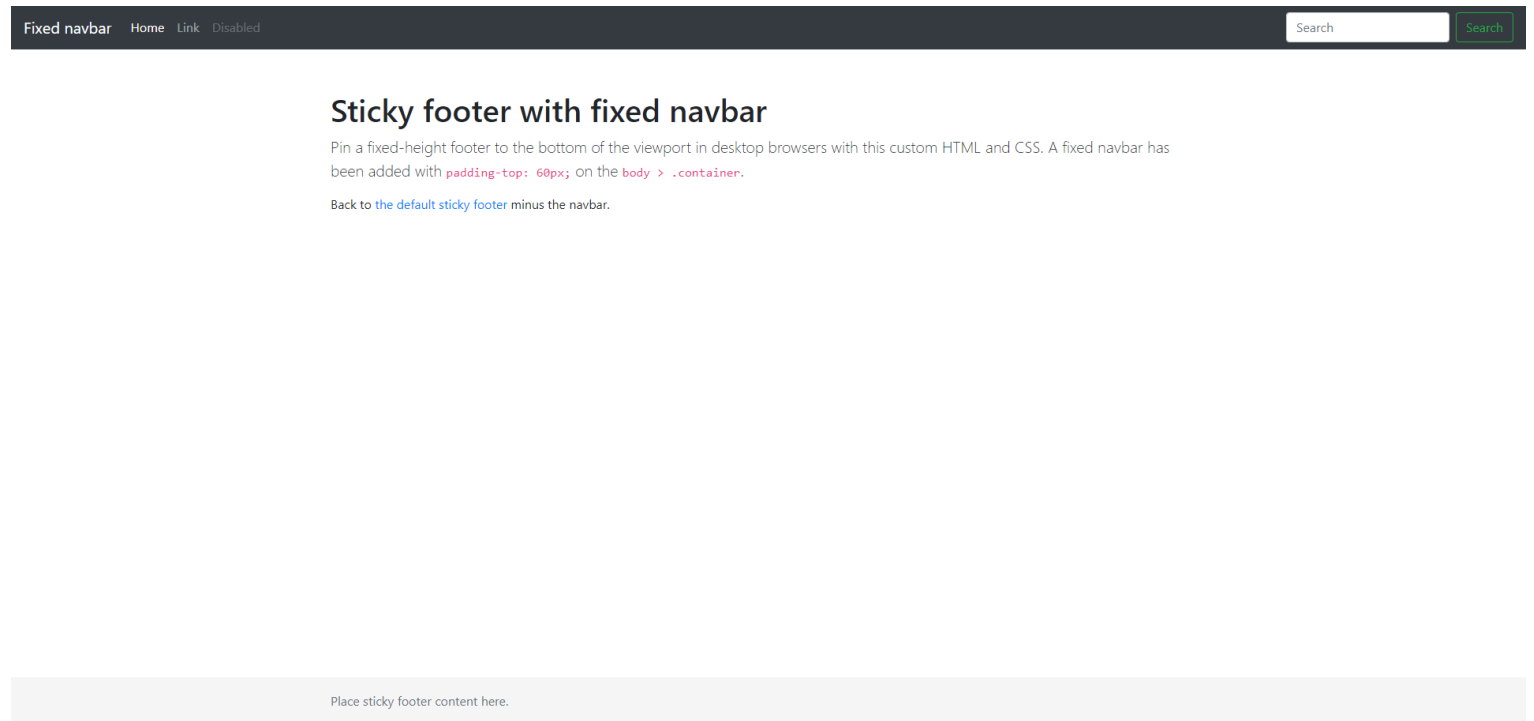
# B

# Przykłady zastosowań



<https://getbootstrap.com/docs/4.3/examples/>

# Przykłady zastosowań





# Jak zacząć używać Bootstrapa?



```
<head>
  <meta charset="utf-8">
  <meta name="viewport" content="width=device-width, initial-scale=1">
  <title>Basic Bootstrap Template</title>
  <!-- Bootstrap CSS -->
  <link href="https://cdn.jsdelivr.net/npm/bootstrap@5.0.2/dist/css/bootstrap.min.css" rel="stylesheet"
    integrity="sha384-EVSTQN3/azprG1Anm3QDgpJLIm9Nao0Yz1ztcQTwFspd3yD65VohhpuuCOmLASjC" crossorigin="anonymous">
</head>
<body>
  <h1>Hello, world!</h1>
  <!-- Bootstrap JS Bundle with Popper -->
  <script src="https://cdn.jsdelivr.net/npm/bootstrap@5.0.2/dist/js/bootstrap.bundle.min.js"
    integrity="sha384-MrcW6ZMFYlzcLA8Nl+NtUVF0sA7MsXsP1UyJoMp4YLEuNSfAP+JcXn/tWtIaxVXM" crossorigin="anonymous"></script>
</body>
</html>
```

<https://www.youtube.com/watch?v=eow125xV5-c>

# Jak zacząć używać Bootstrapa?



```
<head>
  <meta charset="utf-8">
  <meta name="viewport" content="width=device-width, initial-scale=1">
  <title>Basic Bootstrap Template</title>
  <!-- Bootstrap CSS -->
  <link href="https://cdn.jsdelivr.net/npm/bootstrap@5.0.2/dist/css/bootstrap.min.css" rel="stylesheet"
    integrity="sha384-EVSTQN3/azprG1Anm3QDgpJLIm9Nao0Yz1ztcQTwFspd3yD65VohhpuuCOMmLASjC" crossorigin="anonymous">
</head>
<body>
  <h1>Hello, world!</h1>
  <!-- Bootstrap JS Bundle with Popper -->
  <script src="https://cdn.jsdelivr.net/npm/bootstrap@5.0.2/dist/js/bootstrap.bundle.min.js"
    integrity="sha384-MrcW6ZMFYlzcLA8Nl+NtUVF0sA7MsXsP1UyJoMp4YLEuNSfAP+JcXn/tWtIaxVXM" crossorigin="anonymous"></script>
</body>
</html>
```

<https://www.youtube.com/watch?v=eow125xV5-c>



# B

## Przykłady zastosowań

```
<div class="container">
  <h2>Button Styles</h2>
  <button type="button" class="btn">Basic</button>
  <button type="button" class="btn btn-default">Default</button>
  <button type="button" class="btn btn-primary">Primary</button>
  <button type="button" class="btn btn-success">Success</button>
  <button type="button" class="btn btn-info">Info</button>
  <button type="button" class="btn btn-warning">Warning</button>
  <button type="button" class="btn btn-danger">Danger</button>
  <button type="button" class="btn btn-link">Link</button>
</div>
```

### Button Styles

Basic

Default

Primary

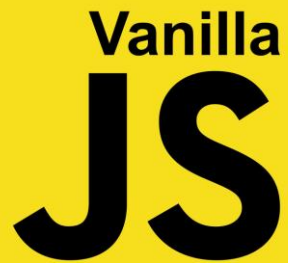
Success

Info

Warning

Danger

Link

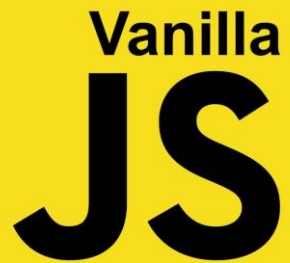
The logo consists of the words "Vanilla" and "JS" in a bold, black, sans-serif font. "Vanilla" is positioned above "JS", which is significantly larger. The text is centered within a bright yellow square. The background of the slide features a stylized illustration of a person with orange hair wearing a pink face mask, sitting at a desk with a laptop. The scene is set against a dark blue background with faint, glowing blue lines and shapes, suggesting a digital or tech environment.

## Vanilla JS

# Vanilla Javascript – Co to jest ?

**JavaScript** to **język programowania**, który stał się jednym z **najpopularniejszych języków programowania** na świecie. Jest to **dynamiczny język programowania**, który pozwala na tworzenie interaktywnych stron internetowych oraz aplikacji webowych. Jako **uniwersalny język programowania**, JavaScript może być używany zarówno na stronie klienta (przeglądarka), jak i serwera (Node.js). **Nowoczesnym języku JavaScript** towarzyszą liczne standardy, takie jak ECMAScript, które wprowadzają nowe funkcje i ułatwienia dla programistów. **Język JavaScript charakteryzuje** się również dynamicznym typowaniem, co oznacza, że typy zmiennych są określane w trakcie działania programu, a nie w trakcie kompilacji.

[https://aviary.pl/programowanie/javascript/#Czym\\_jest\\_JavaScript](https://aviary.pl/programowanie/javascript/#Czym_jest_JavaScript)



# Zmienne w Javascript

## 1. var

- Funkcja-scoped: Zmienna dostępna tylko w obrębie funkcji, nie w blokach.
- Niezalecane ze względu na problem zasięgu.

## 2. let

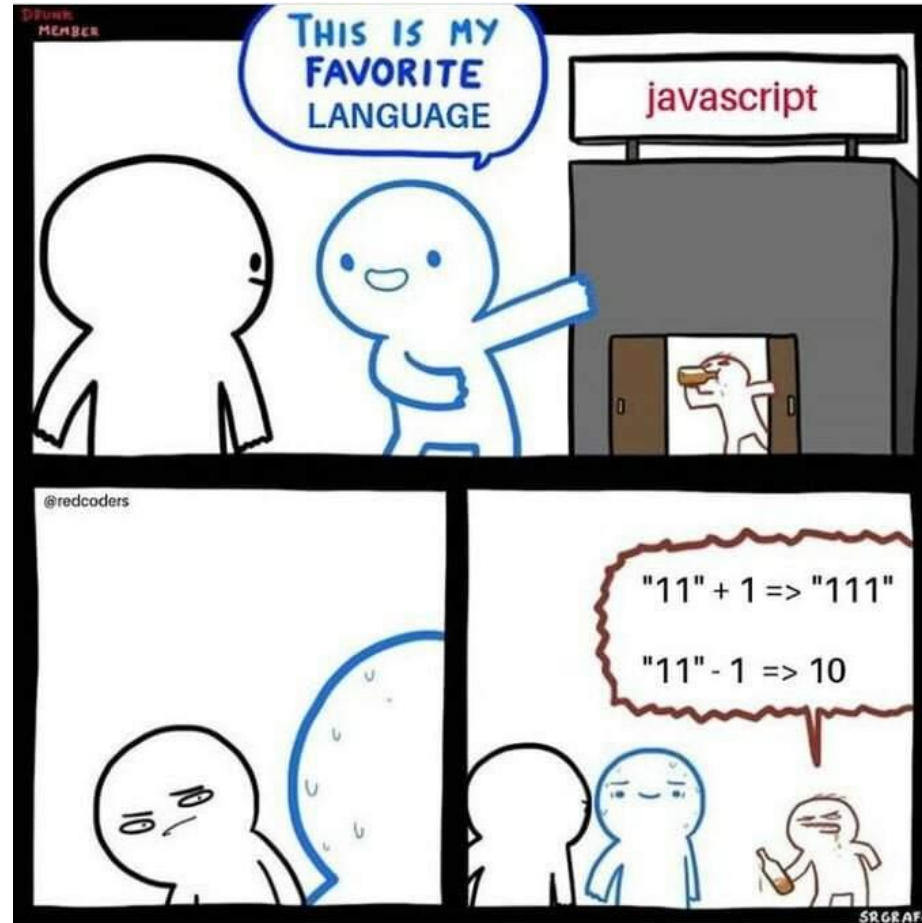
- Blokowo-scoped: Zmienna dostępna tylko w bloku, np. pętli czy warunku.
- Lepszy zasięg, zalecane do użycia.

## 3. const

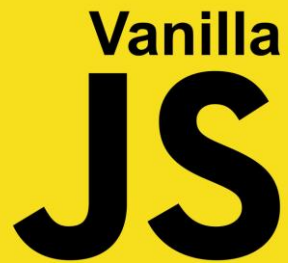
- Blokowo-scoped: Stała, której wartość nie może być zmieniona.
- Bardzo przydatne do deklarowania stałych wartości.

# Wprowadzenie do Javascript

Vanilla  
**JS**





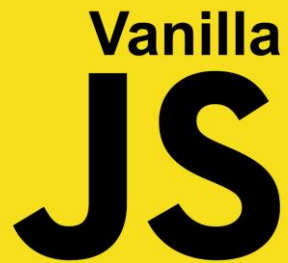
The logo for Vanilla JS, featuring the words "Vanilla" and "JS" in a bold, black, sans-serif font. "Vanilla" is positioned above "JS", which is significantly larger. The text is centered within a bright yellow square. The background of the slide is a dark blue gradient with faint, glowing blue and orange lines and shapes, suggesting a digital or network theme.

## Vanilla JS

# Zalety Javascript

Najważniejsze zalety JavaScript:

- 1.Prostota:** Bardzo intuicyjna składnia umożliwia szybki start w programowaniu po zaledwie kilkunastu minutach nauki.
- 2.Popularność:** JavaScript cieszy się ogromną popularnością, co ułatwia dostęp do poradników i kursów online. Jego efekty można zauważyć praktycznie na każdej stronie internetowej.
- 3.Wpływ na stronę internetową:** Redukuje obciążenie serwera poprzez minimalizowanie ilości przesyłanych danych pomiędzy hostingiem a użytkownikiem. Przykładowo, formularze rejestracyjne mogą być sprawdzane w przeglądarce, eliminując konieczność przesyłania danych, analizy ich na serwerze i odesłania informacji do użytkownika.
- 4.Uniwersalność:** JavaScript umożliwia wykonywanie różnorodnych działań na stronie, począwszy od prostych zmian kolorów czcionek, a skończywszy na zaawansowanych operacjach, sprawiając, że strona staje się interaktywna.
- 5.Ciągły rozwój:** Wbrew przewidywaniom o szybkim upadku JavaScript, język ten rozwija się nadal, utrzymując swoją pozycję i stale ewoluując.

The logo for Vanilla JS, featuring the word "Vanilla" in a smaller, black, sans-serif font above the letters "JS" in a large, bold, black, sans-serif font. The logo is set against a bright yellow square background.


## Vanilla JS

# Wady Javascript

Oto kluczowe wady JavaScript:

- Niejednoznaczna interpretacja kodu przez przeglądarki** – Efekt działania tego samego kawałka kodu może wyglądać zupełnie inaczej w Chrome i Firefoxie. Nawet napisany zgodnie z wszelkimi zaleceniami kod, może w określonych przypadkach zacząć żyć swoim życiem.
- Nieustanny rozwój** – Tak, to także minus. Framework, który stosujemy teraz, może zostać za jakiś czas wyparty przez inny, nowszy, którego trzeba będzie nauczyć się od nowa.
- Nadużywanie JS przez programistów** – JavaScript nie nadaje się do używania wszędzie. W miejscach, gdzie potrzebna jest bardzo wysoka wydajność, lepiej z niego zrezygnować, podczas gdy wiele osób implementuje kod JS także tam, co może powodować komplikacje.

<https://hostclub.pl/javascript/>

The logo consists of the word "Vanilla" in a small, black, sans-serif font, positioned above the letters "JS" in a large, bold, black, sans-serif font. The entire logo is set against a bright yellow square background.

## Vanilla JS

# Przykłady zastosowań

Charakterystyka języka JavaScript umożliwia stworzenie wielu przydatnych funkcjonalności po stronie przeglądarki internetowej. Właśnie dlatego JavaScript stał się niezwykle doceniany przez programistów Front Endu, którzy mogą dzięki niemu stworzyć zarówno niepowtarzalne efekty, jak i wykorzystać JavaScript do zaprogramowania ważnych funkcji. Z jego pomocą możemy stworzyć np. formularz zgłoszeniowy, czy koszyk w sklepie internetowym.



# TS

## Co to jest TypeScript?

TypeScript to język programowania stworzony przez Microsoft w 2012 roku. Jego twórcą jest Anders Hejlsberg, znany również jako ojciec języka C#. **TypeScript jest nadzbiorem JavaScriptu, co oznacza, że każdy poprawny kod JS jest równocześnie poprawnym kodem TS.** W praktyce TypeScript rozszerza możliwości JavaScriptu o opcjonalne statyczne typowanie, nowe struktury danych, takie jak Enumy i Klasy, oraz inne funkcje wymienione w dalszej części artykułu. TS posiada też aktywną społeczność programistów tworzących biblioteki, które ułatwiają pracę programistom poprzez dodawanie typowania do już istniejących projektów JS. Dzięki temu są one przystosowane do pracy z TypeScriptem.

<https://www.droptica.pl/blog/co-jest-typescript-i-dlaczego-sprawdzi-sie-w-twoich-projektach/>

Stylizowane tło z laptopem i kodem. W centralnej części znajduje się biały kwadrat z napisem 'TS' w czarnych literach. W tle, na ciemnoniebieskim tle, widać stylizowane linie i kształty, które mogą być kodem lub elementami interfejsu użytkownika. W lewym górnym rogu jest fragment laptopa z ekranem i klawiaturą. W prawym górnym rogu jest fragment klawiatury. W dolnym lewym rogu jest fragment klawiatury. W dolnym prawym rogu jest fragment klawiatury.

# TS

## Dlaczego warto używać TypeScriptu?

TypeScript zapewnia wiele korzyści, które przekładają się na lepszą jakość kodu, większą produktywność doświadczonych programistów i szybsze tempo pracy. Użycie TS w projektach pozwala na skorzystanie z jego zaawansowanych funkcji i elastyczności, dlatego staje się coraz bardziej popularnym językiem wśród deweloperów.

<https://www.droptica.pl/blog/co-jest-typescript-i-dlaczego-sprawdzi-sie-w-twoich-projektach/>

Stylizowane tło z laptopem i kodem. W centralnej części znajduje się niebieski kwadrat z białymi literami 'TS'. W tle widoczny jest laptop z ekranem, na którym są symbole kodu, oraz stylizowane linie i kształty w kolorach niebieskim i pomarańczowym.

# TS

## Zalety TypeScriptu

- 1. Natychmiastowe wykrywanie błędów typowania:** Dzięki statycznemu typowaniu, TypeScript błyskawicznie identyfikuje błędy typów i nazewnictwa zmiennych.
- 2. Lepsza czytelność kodu:** TypeScript wprowadza interfejsy, klasy i dekoratory, co ułatwia organizację kodu i zwiększa przejrzystość dla innych programistów.
- 3. Łatwiejsze refaktoryzacje:** Statyczne typowanie oraz wsparcie narzędziowe ułatwiają refaktoryzację kodu, zwłaszcza w dużych projektach.
- 4. Integracja z istniejącymi projektami JavaScript:** TypeScript jest kompatybilny z JavaScriptem, umożliwiając stopniowe wdrażanie w istniejących projektach oraz współpracę z bibliotekami i narzędziami JS.
- 5. Wsparcie dla nowoczesnych funkcji języka:** TypeScript pozwala korzystać z najnowszych funkcji ECMAScript nawet w starszych przeglądarkach poprzez transpilację do starszych wersji JavaScriptu.
- 6. Zwiększona produktywność:** TypeScript przyspiesza pisanie kodu dzięki lepszemu wsparciu dla autouzupełniania, nawigacji po kodzie i inteligentnym sugestiom dostępnym w narzędziach deweloperskich.

<https://www.droptica.pl/blog/co-jest-typescript-i-dlaczego-sprawdzi-sie-w-twoich-projektach/>





# Wady TypeScriptu

- 1.Skomplikowany proces nauki:** Wprowadzenie statycznego typowania i innych funkcji może być trudne dla nowych użytkowników.
- 2.Ścisłość i sztywność typowania:** Niektórzy programiści mogą odczuwać ograniczenia wynikające z statycznego typowania.
- 3.Dodatkowy etap kompilacji:** Proces kompilacji dodaje krok do cyklu pracy, co może zwiększyć złożoność projektu.
- 4.Problemy z zgodnością:** Integracja z istniejącymi projektami JavaScript może generować trudności z zgodnością.
- 5.Czas kompilacji w większych projektach:** Wraz ze wzrostem projektu, czas kompilacji może się wydłużać, wpływając na efektywność pracy, zwłaszcza w większych projektach.

<https://www.droptica.pl/blog/co-jest-typescript-i-dlaczego-sprawdzi-sie-w-twoich-projektach/>

Stylizowane tło z laptopem i kodem. W centralnej części znajduje się niebieski kwadrat z białymi literami 'TS'. W tle, na ciemnoniebieskim tle, widać stylizowane linie i symbole, takie jak 'Q', 'I =', '<', oraz '1'.

# TS

# Javascript vs TypeScript

TypeScript różni się od JavaScriptu głównie wspomnianym systemem typów, który pozwala na lepsze zarządzanie i kontrolowanie kodu. Dodatkowo TypeScript **wprowadza takie elementy jak interfejsy, klasy abstrakcyjne czy dekoratory, co sprawia, że kod staje się bardziej przejrzysty i łatwiejszy w utrzymaniu.**

<https://www.droptica.pl/blog/co-jest-typescript-i-dlaczego-sprawdzi-sie-w-twoich-projektach/>



**Pytania**



The logo for Vanilla JavaScript, featuring the word "Vanilla" in a small, black, sans-serif font above the letters "JS" in a large, bold, black, sans-serif font, all contained within a yellow square.

Vanilla  
**JS**

The logo for Babel, featuring a large, white, serif letter "B" centered within a purple, rounded square with a white border.

**B**

The logo for TypeScript, featuring the letters "TS" in a large, white, sans-serif font centered within a blue square.

**TS**

The logo for HTML5, featuring the word "HTML" in a bold, black, sans-serif font above a large, white, stylized number "5" that is partially enclosed by an orange shield-like shape.

**HTML**

# Quiz!