



# Znajdowanie elementów HTML przy pomocy JavaScript

```
let selektor = document.getElementById('id')
```

```
let selektor = document.querySelector('#id')
```



- Metoda ***querySelector()*** zwraca pierwszy element pasujący do selektora CSS
- Aby zwrócić wszystkie dopasowania zamiast tego używamy funkcji ***querySelectorAll()***
- Zarówno ***querySelector()***, jak i ***querySelectorAll()*** zgłaszają wyjątek SYNTAX\_ERR, jeśli selektory są nieprawidłowe

```
let selektor = document.getElementsByTagName('p')
```

```
let selektor = document.querySelectorAll('p')
```



```
let selektor = document.getElementsByClassName('nazwaKlasy')
```

```
let selektor = document.querySelectorAll('.nazwaKlasy')
```





## Różnica między **HTMLCollection** a **NodeList**

Główna różnica między **HTMLCollection** a **NodeList** polega na tym, że jedna jest aktywna, a druga statyczna. Oznacza to, że gdy element jest dołączany do modelu DOM, aktywny węzeł rozpozna nowy element, podczas gdy węzeł statyczny nie.



# HTMLCollection

Metody elementów `getElementsByClassName()` i `getElementsByTagName()` zwracają aktywną kolekcję **HTMLCollection**.

Zawiera ona tylko pasujące elementy (np. nazwę klasy lub nazwę znacznika) i nie zawiera węzłów tekstowych

Udostępnia tylko dwie metody ***item*** i ***namedItem***.



# NodeList

Metoda elementu `*querySelectorAll()*` zwraca statyczną listę węzłów. Wyglądają jak tablice, ale nimi nie są.

NodeLists mają zdefiniowaną metodę `forEach`, a także kilka innych metod, w tym `element`, `wpisy`, `klucze` i `wartości`.

NodeLists zachowują się różnie w zależności od tego, w jaki sposób uzyskujesz do nich dostęp; jeśli uzyskujesz dostęp do elementów za pomocą `childNodes`, zwrócona lista jest aktywna i będzie aktualizowana w miarę dodawania kolejnych elementów do węzła. Jeśli uzyskuje się do niej dostęp za pomocą funkcji `querySelectorAll()`, zwrócona lista jest statyczna i nie będzie aktualizowana, jeśli do węzła zostanie dodanych więcej elementów.