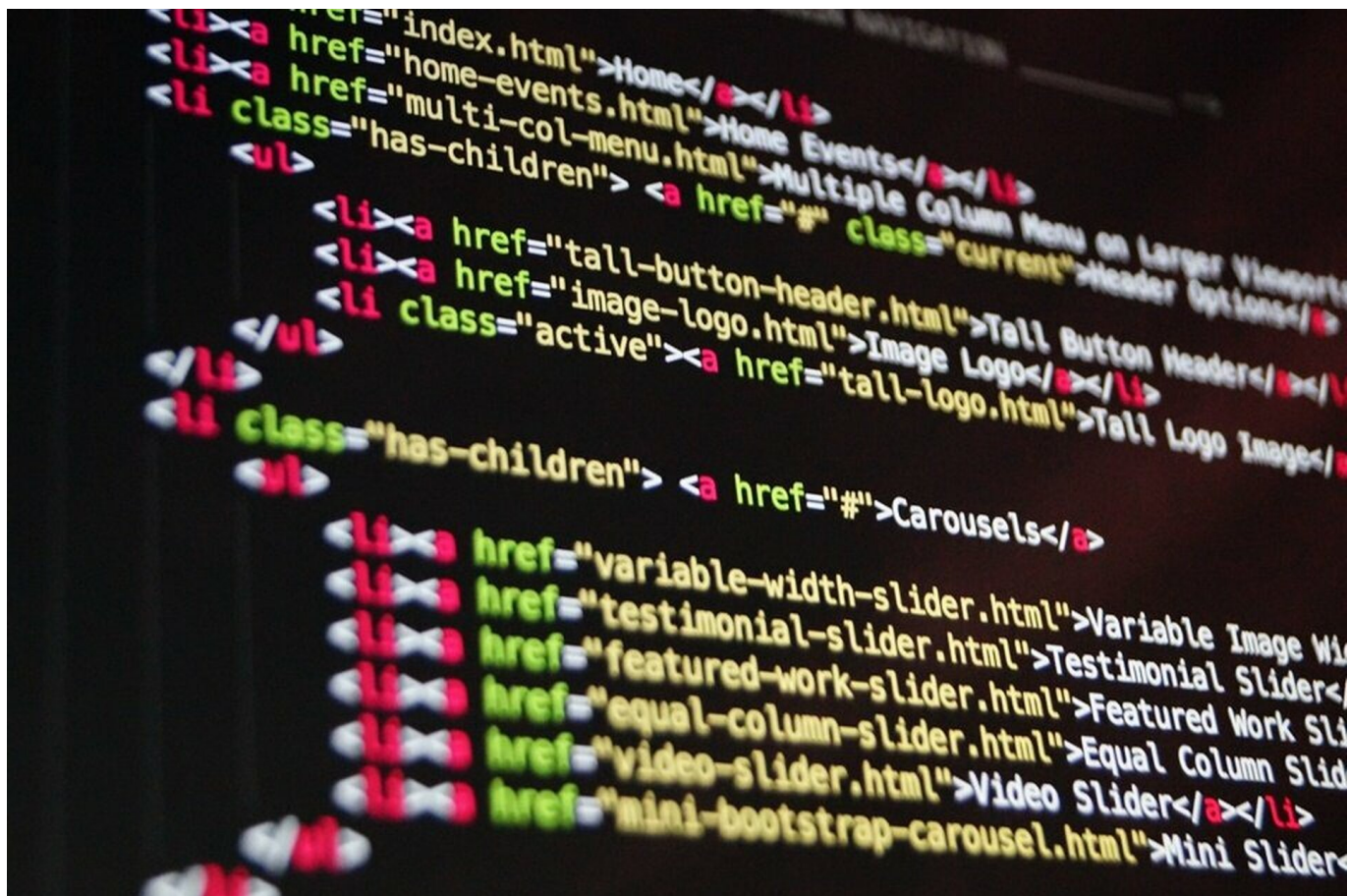




Elementy interaktywne w projekcie strony internetowej

- Wprowadzenie
- Animacja
- Przeczytaj
- Prezentacja multimedialna
- Dla nauczyciela



Elementy interaktywne w projekcie strony internetowej

Źródło: domena publiczna.

Przyjrzyj się stronom internetowym, z których korzystasz. Czy zauważasz elementy, które odpowiadają za to, że witryna komunikuje się z użytkownikiem? A może stały się one zupełnie przezroczyste, już niedostrzegalne dla użytkownika?

Niezależnie od odpowiedzi, przeanalizujemy, w jaki sposób zaimplementować wybrane rozwiązania w budowanej stronie WWW.

Pozostałe e-materiały z serii:

- Dobre praktyki związane z tworzeniem stron internetowych,
- Kreatory stron internetowych,
- Etapy projektowania strony internetowej,
- Fonty i ikony w projekcie strony internetowej,
- Formularze i menu w projekcie strony internetowej,
- Graficzna prezentacja danych na stronie internetowej,
- Nadawanie stylów elementom strony internetowej,
- Tworzenie galerii zdjęć na stronie internetowej,
- Responsywne i mobilne strony internetowe.

Twoje cele

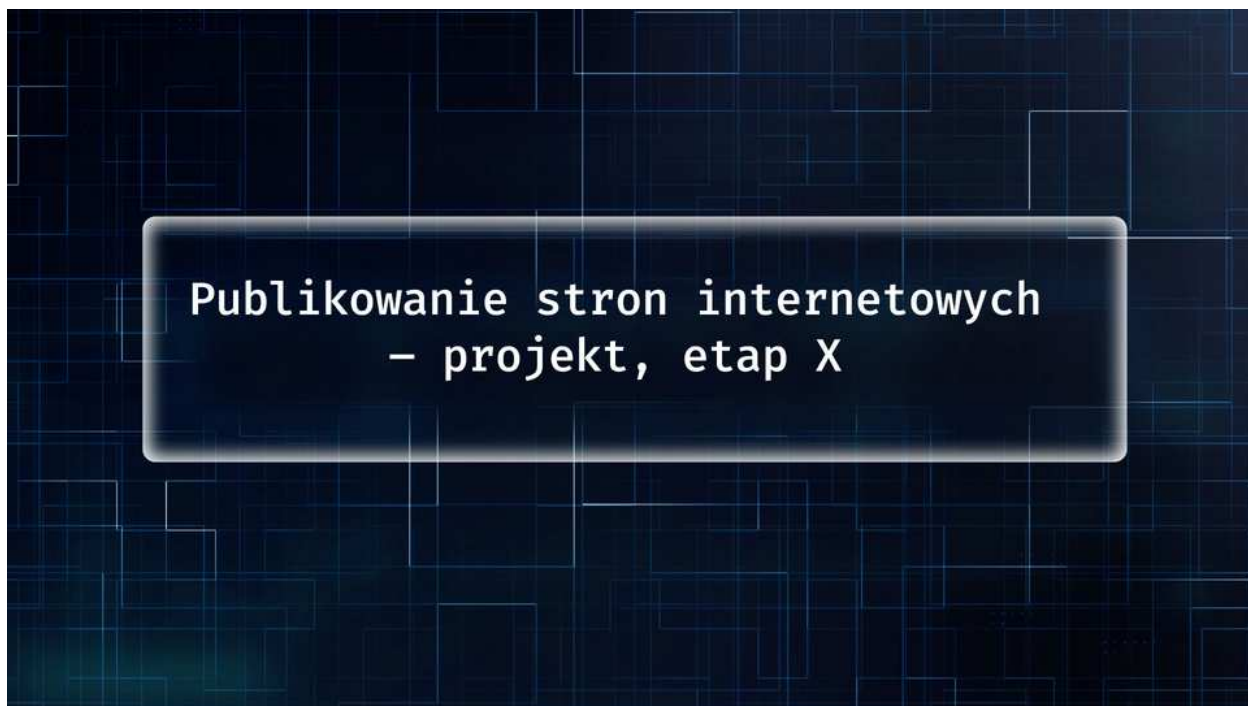
- Wyjaśnisz, na czym polega interaktywność strony WWW.

- Wskażesz elementy interaktywne w wybranej witrynie.
- Zastosujesz język JavaScript lub bibliotekę jQuery, by wprowadzić interaktywność do swojego projektu.

Animacja

Polecenie 1

Zapoznaj się z animacją. Następnie zastanów się, jakie elementy interaktywne są wykorzystywane na odwiedzanych przez ciebie stronach WWW.



Film dostępny pod adresem </preview/resource/RpieYqsjd9OLI>

Źródło: Contentplus.pl Sp. z o.o., licencja: CC BY 3.0.

Film nawiązujący do treści materiału: Publikowanie stron internetowych - projekt, etap X.

Ćwiczenie 1

Zapoznaj się z witryną zpe.gov.pl. Jakie elementy interaktywne zostały wykorzystane w jej budowie?

Ćwiczenie 2

W internecie poszukaj archiwalnej wersji wybranego portalu informacyjnego (np. onet.pl). Jak zmieniały się jego elementy interaktywne? Możesz skorzystać z witryny web.archive.org.

Przeczytaj

Dodaj interaktywność do swojego projektu – JavaScript

Interaktywność stron, wywoływanie zdarzeń po kliknięciu przycisku myszy, walidacja formularzy – to wszystko elementy tworzące na stronie internetowej interakcje z użytkownikiem. Są one obsługiwane głównie dzięki zastosowaniu języków programowania, między innymi języka JavaScript (innym popularnym narzędziem stosowanym w tym celu jest jQuery).

Zacniemy od tego, w jaki sposób załączamy kod JavaScript obsługujący zdarzenia. Następnie wyjaśnimy, jak poruszamy się i znajdujemy kolejne elementy, którym nadajemy interaktywność. Na koniec wykorzystamy wszystkie zdobyte informacje i zaprogramujemy zdarzenie.

Implementacja kodu JavaScript w pliku HTML

Implementacji kodu JavaScript w pliku HTML możemy dokonać na dwa sposoby:

- umieszczając kod w kodzie HTML;
- tworząc osobny plik.

Umieszczenie w kodzie HTML

W ramach pierwszej metody w dolnej części `<body>` umieszczamy `<script>` oraz wstawiamy między znacznikami dowolny kod, który przygotowaliśmy.

```
1 <script>
2     // w tym miejscu wpisujemy kod
3 </script>
```

Przygotowanie w osobnym pliku

W przypadku drugiej metody stworzymy w projekcie plik `script.js` i dodajemy w dokumencie HTML, na końcu `<body>`, tuż przed zamknięciem tagu, źródło pliku JavaScript.

```
1 <script src="./js/script.js" type="text/javascript"></script>
```

Cały kod zawarty w pliku `script.js` jest opakowany w zdarzenie `DOMContentLoaded`, które powoduje, że kod znajdujący się w jego wnętrzu jest wywoływany dopiero w momencie, gdy załaduje się cały HTML. Nie czeka on na zakończenie ładowania arkuszy stylów czy obrazów.

Kiedy mechanika JavaScript jest gotowa do działania, w konsoli deweloperskiej zostaje wyświetlony komunikat `"documents is ready"`.

Wyszukiwanie elementów

Wyszukiwanie elementów na stronie jest kolejnym krokiem do dodania na niej interaktywności. Po ich odnalezieniu możemy dodawać kolejne zdarzenia, które mogą być wywołane, np. po kliknięciu zdjęcia czy przycisku.

Podstawowe metody wyszukiwania elementów oparte na języku JavaScript to:

```
1 document.querySelector("selector");
2     // wyszukuje pierwszy element odpowiadający zapytaniu
3 document.getElementById("id");
4     // wyszukuje element z określonym id
5 document.querySelectorAll("selector");
6     // wyszukuje wszystkie elementy odpowiadające zapytaniu
7 document.getElementsByTagName("tag");
8     // wyszukuje elementy po tagu
9 document.getElementsByClassName("className");
10    // wyszukuje elementy po nazwie klasy
```

Przykładowo, jeżeli chcemy zaleźć wszystkie paragrafy zawarte w znacznikach `div` w strukturze naszej strony, w pierwszej kolejności musimy zdefiniować zmienną, w której wywołujemy odpowiednią metodę.

```
1 var paragraf = document.querySelectorAll("div p");
```

Aby sprawdzić poprawność działania naszego kodu, dodajemy `console.log`, który wyświetli nam w konsoli deweloperskiej informacje.

```
1 var paragraf = document.querySelectorAll("div p");
2 console.log(paragraf);
```

Polecenie 1

Korzystając z [przykładowego projektu strony WWW](#), sprawdź możliwość wyszukiwania elementów. Otwórz plik `script.js` i na podstawie przedstawionych powyżej metod wykonaj następujące czynności:

1. Znajdź pierwszy element `h2` na stronie.
2. Znajdź element o klasie `welcome`.
3. Znajdź element o id `chartdiv`.
4. Znajdź wszystkie elementy `li`, znajdujące się w tagu `ul`.

Pamiętaj, aby zdefiniować zmienne i zastosować `console.log`, który w konsoli deweloperskiej pozwoli sprawdzić poprawność kodu.

Skoro HTML zbudowany jest na kształt rodziny, w której elementy mają rodziców, a często także rodzeństwo i dzieci, możemy również – poprzez relacje między nimi – wyszukiwać poszczególne elementy.

Znajdźmy pierwszy element o klasie `"card-description"`, a następnie wypiszmy w konsoli jego rodzica.

```
1 var card = document.querySelector(".card-description");  
2 console.log(card.parentElement);
```

Polecenie 2

Analogicznie wyszukaj element o klasie `"card"`, a następnie – korzystając ze słowa `"children"` – wypisz w konsoli wszystkie jego dzieci.

Tablice w JavaScript

Zgodnie z rozwiązaniem uzyskanym w **Poleceniu 2** możemy zauważyć, że otrzymaliśmy zbiór różnych wartości. Jest to tablica, a jej konstrukcja wygląda następująco:

Tablica z czterema elementami:


```
1 var array = [1, 2, 3, "czwarty element"];
```

Pusta tablica:

```
1 var arrayEmpty = [];
```

Tablice indeksujemy od zera, zatem pierwszy element tablicy ma indeks 0.

```
1 console.log(array[0]);  
2 // wyświetli wartość 1
```

Aby sprawdzić wielkość tablicy, możemy skorzystać z atrybutu `length`.

```
1 console.log(array.length);  
2 //wyświetli 4
```

Polecenie 3

W pliku `j s` stwórz tablicę z pięcioma różnymi owocami, a następnie wypisz w konsoli deweloperskiej, ile znajduje się tam elementów, oraz podaj nazwy trzeciej i szóstej wartości.

Wywoływanie zdarzeń

Eventy to zdarzenia wywoływane na [stronie WWW](#). Możemy je wywoływać w zależności od rodzaju interakcji, np. poprzez kliknięcie, najechanie kursorem na element, przy edycji formularza, za pomocą klawiatury czy myszy. Aby dodać zdarzenie na stronie, w pierwszej kolejności wybieramy element, na jakim ma się ono zrealizować. U nas jest to przycisk o `id = submitBtn`

```
1 var button = document.querySelector("#submitBtn");
```

Na przycisku dodajemy metodę `addEventListener()`. Podstawowe parametry, jakie wskazujemy w tej funkcji, określają, w jakiej sytuacji wydarzenie ma się wykonać (u nas przykładowo będzie to `"click"`) i co dokładnie ma się stać po jego wywołaniu.

```
1 var button = document.querySelector("#submitBtn");
```

```
2
3 button.addEventListener("click", function(event) {
4     // po kliknięciu w konsoli deweloperskiej pojawi się tekst "b
5     console.log("button test");
6 });
```

Podsumowując: zdefiniowaliśmy zdarzenie, które jest wywoływane na przycisku o `id = submitBtn`. Po jego kliknięciu, w konsoli dewelopera wyświetli się tekst: "button test".

Zobaczmy jeszcze kolejny przykład działania kodu w języku JavaScript.

Zdarzenie dotyczy wyświetlenia przepisu po kliknięciu przycisku „Zobacz mój ostatni przepis”. W pierwszej kolejności musimy znaleźć zatem ten element i przypisać go do zmiennej. Od razu sprawdzimy, czy kod, który zapisaliśmy, jest poprawny, przez użycie `console.log`.

```
1 var recipe = document.getElementsByClassName("show-recipe");
2 console.log(recipe);
```

Po wywołaniu zmiennej w konsoli deweloperskiej widzimy, że to tablica.

Musimy zatem pamiętać, aby przy wywoływaniu zdarzenia wskazać, że jest to pierwszy element - robimy to poprzez dodanie `[0]`.

```
1 var recipe = document.getElementsByClassName("show-recipe");
2
3 recipe[0].addEventListener("click", function() {
4     // ciało funkcji
5 });
```

Po kliknięciu przycisku „Zobacz mój ostatni przepis” chcemy, aby pojawił się przepis. Następnym krokiem jest w takim razie odnalezienie elementu o klasie `.favorite-recipe` (który ma się wyświetlać lub chować po kliknięciu przycisku) oraz skorzystanie z konstrukcji warunkowej `if`. Konstrukcję tę znamy doskonale z innych języków programowania, więc nie będziemy jej omawiać w tym miejscu.

```
1 var recipe = document.getElementsByClassName("show-recipe");
2
3 recipe[0].addEventListener("click", function(e) {
```

```
4     var parentEl = this.parentElement.parentElement;
5     var favoriteRecipe = parentEl.getElementsByClassName("favorit
6
7     if (favoriteRecipe[0].style.display === "flex") {
8         favoriteRecipe[0].style.display = "none";
9     } else {
10         favoriteRecipe[0].style.display = "flex";
11     }
12
13 });
```

Ciekawostka

Wyszukiwanie elementów w dokumencie HTML może również być związane z relacjami względem innych elementów, np. rodzic, dziecko, rodzeństwo. W przypadku omówionego wyżej zdarzenia odnaleźliśmy element odnosząc się do relacji dziecko-rodzic, wykorzystując sformułowanie `parentElement`.

Polecenie 4

Na podstawie przeanalizowanego kodu spróbuj wykorzystać informację dotyczącą wywoływania zdarzeń na stronie internetowej i dodaj kod tak, aby po kliknięciu przycisku „Zobacz mój ostatni przepis”, podświetlał się on na czerwono. Kiedy przepis zostanie schowany, należy przywrócić pierwotny kolor.

Polecenie 5

Znajdź w repozytorium element o klasie `color_photo`, a następnie utwórz zdarzenie, które – po podwójnym kliknięciu zdjęcia – będzie w konsoli wypisywało `dblclick`.

Słownik

dokument HTML

dokument (strona WWW) czytany przez przeglądarki internetowe, widoczny pod konkretnym adresem URL; zawiera tekst, odnośniki (linki), informacje o multimediami i zdjęciach

link

inaczej odnośnik; aktywny fragment tekstu albo obrazek; pozwala przejść na inny adres URL albo wykonać inną czynność, np. przesłać zapytanie przez formularz kontaktowy

strona internetowa

zbiór dokumentów HTML połączonych ze sobą linkami URL (odnośnikami); dokumenty te znajdują się w obrębie jednej domeny internetowej (serwisu albo witryny)

tablica

rodzaj danych, który pozwala w jednej zmiennej przechowywać wiele wartości; może zawierać różne typy danych, m.in. liczby, stringi, obiekty, funkcje

Prezentacja multimedialna

Polecenie 1

Zapoznaj się z prezentacją multimedialną, która wyjaśnia kolejno:

- jak załączyć jQuery do pliku HTML,
- jak wyszukiwać elementy, na których potem będziemy mogli wywoływać zdarzenia.

Materiał audio dostępny pod adresem:

<https://zpe.gov.pl/b/PxpFDk7cz>

Dodaj interaktywność do swojego projektu,
używając jQuery

1

2

Materiał audio dostępny pod adresem:

<https://zpe.gov.pl/b/PxpFDk7cz>

Dodanie jQuery do pliku HTML

Biblioteka jQuery napisana jest w języku
JavaScript. Służy łatwiejszemu wyszukiwaniu
elementów oraz używaniu eventów.

3

Materiał audio dostępny pod adresem:

<https://zpe.gov.pl/b/PxpFDk7cz>

Zastosowanie jQuery na stronie internetowej jest możliwe poprzez pobranie plików ze strony projektu i dodanie do biblioteki za pomocą tagu `<script>` lub skorzystanie z plików umieszczonych już w innym projekcie opublikowanym w internecie.

4

Materiał audio dostępny pod adresem:

<https://zpe.gov.pl/b/PxpFDk7cz>

W projekcie korzystamy z drugiego rozwiązania – implementacja wygląda następująco:

Aby zrozumieć konstrukcję jQuery, przeanalizujemy kilka następnych kroków.

5

Materiał audio dostępny pod adresem:

<https://zpe.gov.pl/b/PxpFDk7cz>

Wyszukiwanie elementów

jQuery pozwala na wyszukiwanie elementów za pomocą selektorów CSS. Przykładowe użycie tej biblioteki wygląda następująco:

6

Materiał audio dostępny pod adresem:

<https://zpe.gov.pl/b/PxpFDk7cz>

Do wyszukiwania elementów możemy wykorzystywać `find()`, `children()`, `first()`, `last()`, `eq()`.

Polecenie 2

Korzystając z [przykładowego projektu strony WWW](#), na podstawie przedstawionych informacji sprawdź możliwość wyszukania elementów. Otwórz plik `script.js` i wykonaj poniższe polecenia. Pamiętaj, aby zapisać metody do zmiennych i wywołać je, używając `console.log` (analogicznie, jak robiliśmy to w przypadku JavaScript).

- Korzystając z `next()`, znajdź element o id `Welcome` i zwróć następny element.
- Korzystając z `prev()`, znajdź elementy o klasie `menu`, a następnie zwróć poprzedni.
- Korzystając z `siblings()`, znajdź `label` i zwróć rodzeństwo każdego z nich.

Polecenie 3

Zapoznaj się z prezentacją, w której omówiono dodawanie zdarzeń przy użyciu jQuery. Następnie wykonaj polecenie.

Materiał audio dostępny pod adresem:

<https://zpe.gov.pl/b/PxpFDk7cz>

Dodawanie zdarzeń

Przeanalizujmy przykładowe zdarzenie dotyczące scrollowania w odpowiednie miejsce strony (z wyznaczoną szybkością) po kliknięciu wybranego elementu z menu.

1



Materiał audio dostępny pod adresem:

<https://zpe.gov.pl/b/PxpFDk7cz>

Aby wywołać zdarzenie, zamiast znanej w JavaScript metody `addEventListener()`, użyjemy konstrukcji `on(event, function)`. Zdarzenie jest związane z konkretnym elementem w całym DOMie HTML, dlatego na początku wskazujemy go. Wyjściowo nasłuchiwanie jest ustawione w całym dokumencie, a dopiero później precyzyjnie wskażemy, na jakich jego elementach ma działać.

|

Materiał audio dostępny pod adresem:

<https://zpe.gov.pl/b/PxpFDk7cz>

3

Rodzaj eventu zależy od typu reakcji użytkownika, która ma być nasłuchiwana na wybranym elemencie. Może to być kliknięcie na element, wybranie określonego klawisza z klawiatury czy inne. W analizowanym przykładzie jest to `"click"`.

|

4

Materiał audio dostępny pod adresem:

<https://zpe.gov.pl/b/PxpFDk7cz>

Naszym celem jest, aby po kliknięciu na element z menu użytkownik został przekierowany do odpowiedniej części na stronie. Nie chcemy jednak, żeby strona była ponownie przeładowana, dlatego skorzystamy z metody `preventDefault()`.

Materiał audio dostępny pod adresem:

<https://zpe.gov.pl/b/PxpFDk7cz>

Dodając animację do scrollowania w projekcie, wprowadzamy dodatkowy parametr. Jest nim dokładne wskazanie elementów, na których ma być wywoływane zdarzenie – czyli linki.

```
//animacja przy scrollowaniu
$(document).on("click", "a", function(e){
    e.preventDefault();
    var id = $(this).attr("href"),
        topSpace = 10;
    //alert(id);
    $('html, body').animate({
        scrollTop: $(id).offset().top - topSpace
    }, 1000);
});
```

Materiał audio dostępny pod adresem:

<https://zpe.gov.pl/b/PxpFDk7cz>

Podsumujmy zapis w ciele funkcji. Akcja następuje po kliknięciu na każdy element <a> w dokumencie. Dochodzi do zablokowania przeładowania strony i zscrollowania jej w miejsce odpowiadające wartości atrybutu href – zatem funkcja znajduje element, który ma id identyczne jak href klikniętego linku. Całość strony jest scrollowana do wybranego miejsca w czasie jednej sekundy, z uwzględnieniem niewielkiego marginesu w górnej części.

Polecenie 4

Przekształć zdarzenie, które zostało utworzone przy użyciu JavaScript w poleceniu nr 5 w sekcji „Przeczytaj”: znajdź w repozytorium element o klasie `color_photo`, a następnie utwórz zdarzenie, które – po podwójnym kliknięciu w to zdjęcie – będzie w konsoli wypisywało „dblclick”.

Dla nauczyciela

Autorka: Paulina Wierzbińska

Przedmiot: Informatyka

Temat: Elementy interaktywne w projekcie strony internetowej

Grupa docelowa:

III etap edukacyjny, liceum ogólnokształcące, technikum

Podstawa programowa:

Cele kształcenia – wymagania ogólne

I. Rozumienie, analizowanie i rozwiązywanie problemów na bazie logicznego i abstrakcyjnego myślenia, myślenia algorytmicznego i sposobów reprezentowania informacji.

III. Posługiwanie się komputerem, urządzeniami cyfrowymi i sieciami komputerowymi, w tym: znajomość zasad działania urządzeń cyfrowych i sieci komputerowych oraz wykonywania obliczeń i programów.

V. Przestrzeganie prawa i zasad bezpieczeństwa. Respektowanie prywatności informacji i ochrony danych, praw własności intelektualnej, etykiety w komunikacji i norm współżycia społecznego, ocena zagrożeń związanych z technologią i ich uwzględnienie dla bezpieczeństwa swojego i innych.

Treści nauczania – wymagania szczegółowe

I. Rozumienie, analizowanie i rozwiązywanie problemów.

Zakres rozszerzony. Uczeń spełnia wymagania określone dla zakresu podstawowego, a ponadto:

5) przedstawia sposoby reprezentowania w komputerze znaków, liczb, wartości logicznych, obrazów, dźwięków, animacji;

II. Programowanie i rozwiązywanie problemów z wykorzystaniem komputera i innych urządzeń cyfrowych.

Zakres podstawowy. Uczeń:

3) przygotowuje opracowania rozwiązań problemów, posługując się wybranymi aplikacjami:

f) tworzy stronę internetową zgodnie ze standardami, wzbogaconą tabelami, listami, elementami dynamicznymi, posługuje się arkuszem stylów, korzysta z oprogramowania i serwisów przeznaczonych do tworzenia stron; potrafi opublikować własną stronę w internecie;

IV. Rozwijanie kompetencji społecznych.

Zakres rozszerzony. Uczeń spełnia wymagania określone dla zakresu podstawowego, a ponadto:

2) analizuje i charakteryzuje wpływ trendów w historycznym rozwoju pojęć, metod informatyki oraz technologii na możliwości rozwiązywania problemów teoretycznych i praktycznych;

Kształtowane kompetencje kluczowe:

- kompetencje cyfrowe;
- kompetencje osobiste, społeczne i w zakresie umiejętności uczenia się;
- kompetencje matematyczne oraz kompetencje w zakresie nauk przyrodniczych, technologii i inżynierii.

Cele operacyjne (językiem ucznia):

- Wyjaśnisz, na czym polega interaktywność strony WWW.
- Wskażesz elementy interaktywne w wybranej witrynie.
- Zastosujesz język JavaScript lub bibliotekę jQuery, by wprowadzić interaktywność do swojego projektu.

Strategie nauczania:

- konstruktywizm;
- konektywizm.

Metody i techniki nauczania:

- dyskusja;
- rozmowa nauczająca z wykorzystaniem multimediu i ćwiczeń interaktywnych.

Formy pracy:

- praca indywidualna;
- praca w parach;
- praca w grupach;
- praca całego zespołu klasowego.

Środki dydaktyczne:

- komputery z głośnikami, słuchawkami i dostępem do internetu;
- zasoby multimedialne zawarte w e-materiale;
- tablica interaktywna/tablica, pisak/kreda.

Przebieg lekcji

Przed lekcją:

1. Uczniowie, korzystając z innych e-materiałów, przypominają sobie najważniejsze informacje dot. projektowania witryn WWW.
2. Uczniowie przygotowują przykłady witryn, które wykorzystują interaktywność w ciekawy sposób (przykłady: hkdi.edu.hk, ispy.heihei.resn.co, tarotobot.illo.tv).

Faza wstępna:

1. Chętna lub wybrana osoba intuicyjnie wyjaśnia, co oznacza, że strona jest interaktywna. Pozostałe osoby odnoszą się do jej wypowiedzi.
2. Uczniowie dyskutują na temat tego, jak zmieniała się przez lata interaktywność stron WWW.

Faza realizacyjna:

1. Uczniowie zapoznają się z treścią e-materiału. Nauczyciel w razie potrzeby wyjaśnia niezrozumiałe treści.
2. Uczniowie w parach powtarzają przedstawione w e-materiale działania.
3. Uczniowie po zapoznaniu się z animacją przedstawiają przykłady interesującego wykorzystania interaktywności.
4. Nauczyciel inicjuje dyskusję o tym, jak w przyszłości mogą interaktywnością operować strony WWW.

Faza podsumowująca:

1. Uczniowie zapoznają się z prezentacją i wykonują polecenie.

Praca domowa:

1. Wykonaj wszystkie przedstawione w e-materiale zadania, ale już dla twojej właściwej witryny.

Wskazówki metodyczne:

- Nauczyciel może zdecydować, czy uczniowie realizują zadania indywidualnie, czy w grupach.