

Tekst, listy i odnośniki — mini serwis

Czas: 2 × 45 min

praca indywidualna

Cel: Zbudowanie serwisu z trzech podstron połączonych menu, z poprawnie dobranymi znacznikami tekstu, listami zagnieżdżonymi oraz różnymi rodzajami odnośników.

Po zajęciach potrafisz:

- dobiera znaczniki tekstu (strong, em, mark, q, blockquote, br) do znaczenia treści
- tworzy listy uporządkowane, nieuporządkowane, zagnieżdżone i listę definicji
- stosuje ścieżki względne do plików w podkatalogach i katalogu nadrzędnym (../)
- tworzy kotwice wewnątrz długiej strony oraz odnośniki mailto i tel
- otwiera odnośniki zewnętrzne w nowej karcie z atrybutem rel

Sprzęt i materiały

- komputer z Windows 11
- pliki `r1_03_tresci.txt` i `r1_03_rower.svg` z folderu Pliki_do_zadan

Oprogramowanie

- **Visual Studio Code** — rozszerzenie Live Server (autor: Ritwick Dey) — instalacja z panelu Rozszerzenia (Ctrl+Shift+X) code.visualstudio.com
- **Przeglądarka Chrome, Edge lub Firefox** — narzędzia deweloperskie — klawisz F12
- **Walidator W3C (Nu Html Checker)** — sprawdzanie przez wklejenie kodu (text input) lub wgranie pliku (file upload) validator.w3.org/nu

Bezpieczeństwo

- Odnośniki zewnętrzne otwierane w nowej karcie (`target="_blank"`) opatruj atrybutem `rel="noopener noreferrer"` — nowa strona nie dostaje wtedy dostępu do karty, z której ją otwarto.
- W prawdziwym serwisie nie publikuj prywatnych adresów e-mail i numerów telefonów uczniów.

Przebieg zadania

1 Struktura serwisu

Serwis ma trzy podstrony. Plik `trasy.html` leży w podkatalogu — zwróć uwagę na ścieżki.

```
zad03/  
├── index.html  
├── kontakt.html  
├── img/  
│   └── rower.svg  
└── trasy/  
    └── trasy.html
```

1. Utwórz foldery i puste pliki według schematu. Plik `r1_03_rower.svg` zapisz jako `img/rower.svg`.
2. W każdym pliku wpisz szkielet HTML5 (`lang="pl"` , własny `title`) z elementami `header` , `nav` , `main` i `footer` . W stopce podaj imię, nazwisko i klasę, rozdzielone znacznikiem `br` .
3. W `nav` każdej strony umieść listę `ul` z trzema odnośnikami: Strona główna, Trasy, Kontakt.
Uwaga: w `trasy/trasy.html` ścieżki muszą zaczynać się od `../` (np. `../index.html`), a w plikach głównych — nie.
4. W nagłówku `index.html` i `trasy/trasy.html` wstaw rysunek roweru (`width="200"` `height="120"` , sensowny `alt`). Jaka będzie ścieżka do obrazu w każdym z tych plików?

2 Strona główna — tekst i listy

Treść skopiuj z pliku `r1_03_tresci.txt` .

1. Wyróżnij `strong` słowa „każdy uczeń szkoły”, hasło klubu wstaw w znacznik `q` (cudzysłowy doda przeglądarka — nie wpisuj ich), słowo „Najważniejsze” oznacz `mark` , a słowo „zawsze” — `em` .
2. Plan sezonu zapisz jako listę `ul` zagnieżdżoną na dwóch poziomach (pora roku → wydarzenia). Pamiętaj: lista wewnętrzna leży wewnątrz `li` , przed jego znacznikiem zamykającym.
3. Słowniczek zapisz listą definicji `dl` / `dt` / `dd` .
4. Słowa patrona klubu umieść w `blockquote` , a autora cytatu w osobnym akapicie pod cytatem.

3 Długa strona z kotwicami

Kotwica to odnośnik do elementu o danym `id` na tej samej (`#id`) lub innej stronie (`plik.html#id`).

1. W `trasy/trasy.html` opisz pięć tras — każda w osobnej `section` z `id` (`trasa1` ... `trasa5`) i nagłówkiem `h2` . Długość trasy wyróżnij `strong` .
2. Na początku `main` utwórz spis treści jako listę `ol` z odnośnikami `#trasa1` ... oraz do sekcji z ekipunkiem.
3. Nadaj `header` identyfikator `gora` i pod każdą sekcją dodaj odnośnik „Do góry”. Zmniejsz okno przeglądarki i sprawdź działanie wszystkich kotwic.
4. Listę rzeczy do zabrania zapisz jako `ol` . Dodaj odnośnik do OpenStreetMap otwierany w nowej karcie z `rel="noopener noreferrer"` .
5. Na stronie głównej dodaj zdanie z odnośnikiem prowadzącym bezpośrednio do opisu trasy nad jezioro (plik i kotwica w jednym adresie).

4 Kontakt i sprawdzenie

1. W `kontakt.html` dodaj odnośnik `mailto:` z tematem wiadomości „Zapis do klubu” (spację w adresie zapisz jako `%20`) oraz odnośnik `tel:` (numer bez spacji, z `+48`). Odnośnik do PTTK otwórz w nowej karcie.
2. Przejdź cały serwis wyłącznie przez menu i kotwice. W F12 → Network sprawdź, czy żaden plik nie ma statusu 404.
3. Sprawdź wszystkie trzy pliki walidatorem W3C i popraw błędy.

W sprawozdaniu umieść

- ☐ zrzut struktury folderu `zad03` w VS Code
- ☐ tabela odnośników: skąd → dokąd, wpisana ścieżka, rodzaj (względny, kotwica, zewnętrzny, mailto, tel)
- ☐ zrzut strony głównej z listą zagnieżdżoną i słowniczkiem
- ☐ wynik walidacji trzech plików (0 błędów)
- ☐ odpowiedzi na pytania kontrolne

Pytania kontrolne

1. Czym różni się ścieżka względna `img/rower.svg` od `../img/rower.svg` i od `/img/rower.svg`?
2. Kiedy użyjesz `strong`, a kiedy `mark`? Dlaczego nie warto używać `b` tylko po to, by pogrubić tekst?
3. Do czego służy `rel="noopener noreferrer"` przy odnośniku z `target="_blank"`?
4. Co się stanie po kliknięciu odnośnika `mailto:` na komputerze bez skonfigurowanego programu pocztowego?

Kryteria oceny

Kryterium	Punkty
Struktura folderów i działające menu na 3 stronach (poprawne <code>../</code> w podkatalogu), obraz widoczny na obu stronach	4
Znaczniki tekstu: strong, em, mark, q, blockquote, br użyte zgodnie z treścią	3
Lista zagnieżdżona (2 poziomy), lista ol i lista definicji dl	3
Spis treści z 6 kotwicami, odnośniki „Do góry”, odnośnik plik + kotwica ze strony głównej	3
mailto z tematem, tel, dwa odnośniki zewnętrzne z target i rel	2
Walidacja bez błędów, brak 404 w Network, tabela odnośników	2
Odpowiedzi na pytania kontrolne	3
Razem	20

Sprawozdanie przygotuj na wzorze dołączonym do zadania (plik .docx o tym samym numerze).