

## Zadanie nr 1 – Layout, RWD

Stwórz responsywną stronę o układzie jak na załączonych rysunkach. Strona musi zawierać sekcje: header, sidebar 1, sidebar 2, content

W content mają być zakładki.

Wytyczne do wersji mobilnej:

- a) ukryj sidebar 1
- b) zakładki mają przyjąć postać rozwijalnego menu (UWAGA!! to muszą być te same elementy html – nie dopuszczalne jest robienie osobnych elementów dla wersji mobilnej i osobnej dla szerokich ekranów)
- c) czcionka: sans-serif, 18px
- d) każda zakładka ma mieć własną treść (wystraczy pole w innym kolorze z tekstem typu „treść zakładka 1”, „treść zakładka 2” etc. Klikając w daną zakładkę ma się pokazać odpowiadająca jej treść.
- e) kod javascript oraz css ma być zawarty w oddzielnych plikach i tylko załączony do pliku index.php W index.php umieścić cały html

Mile widziane użycie Tw Bootstrap 3 (ale nie wymagane). Nie ma konieczności robienia wszystkich rozdzielczości – ważne, aby rezultat wyglądał jak najbardziej podobnie do screenów na tabletach i monitorach o rozd. do 1280px.

## Zadanie nr 2 - ajax

Jedną z zakładek nazwij : „Rozkład jazdy”

W treści tej zakładki w prawym górnym rogu dodaj klawisz odświeżania. Dodaj także pole typu input.

Następnie za pomocą javascript (może być też jquery) po naciśnięciu klawisza odświeżania ma zostać załadowana z serwera treść , która zostanie następnie pokazana w formie tabelarycznej. Przykładowa treść może być taka sama jak na załączonym rysunku.

Request na serwer ma iść jako POST z jednym parametrem o nazwie „search\_input”

Wartość tego parametru ma być brana na podstawie tego, co zostanie wpisane do inputa (dodanego na stronie obok klawisza odświeżania)

Kod odpowiedzialny za odpowiedź ma zostać umieszczony w pliku o nazwie search.php

## Zadanie nr 3

Utwórz w bazie danych mysql strukturę pozwalającą na trzymanie rozkładu jazdy autobusów. Dane który musi trzymać ta struktura to

- a) nazwa autobusu
- b) godziny odjazdów
- c) miejscowości do których jedzie (trasa – czyli nie tylko miejscowość docelowa, ale także pośrednie)

Dane testowe masz przedstawione na screenie w zadaniu 2 – użyj tych samych

Wykonując to zadanie zwróć uwagę na zadanie 4

#### Zadanie 4

Zmień akcję wykonaną w zadaniu 2 tak, aby po wpisaniu jakiegos tekstu zwracało listę pojazdów i ich godziny odjazdu ale tylko tych w których tekst pasuje do miejscowości przez którą dany pojazd przejeżdża.

Dodaj także pole do wyboru godziny odjazdu (ma ono również isc jako parametr w wywołaniu ajax) i być dodatkowym filtrem. Dane pobierz z bazy stworzonej w zadaniu 3. Do pobierania danych wykorzystaj PDO i pisz zapytania odporne na sql injection. A dane pokazuj w sposób odporny na ataki xss.

Założenie 1) – w inputcie nie trzeba wpisywać całej nazwy miejscowości – wystarczy wpisać kilka znaków i jeśli pasuje to dany autobus powinien zostać wyświetlony. Czyli jak wpiszę „wro” to autobus jadący do wrocławia powinien się pokazać. Jak wpiszę „łó”, to powinny się pokazać dwa autobusy które jadą przez Łódź

Założenie 2) jeśli nic nie zostało wpisane – pokazać wszystkie autobusy

Założenie 3) Jeśli wybiorę godzinę 14:30 to pokazać autobusy odjeżdżające po tej godzinie, czyli

Godzina	Miejscowość	Przez
12:30	Warszawa	(przez: Kalisz, Łódź)
14:14	Wrocław	(przez: Leszno, Rawicz)
14:30	Koszalin	(przez: Oborniki, Piła)
16:10	Karków	(przez: Łódź, Katowice)

Oczywiście filtry można łączyć czyli wybrać godzinę oraz wpisać jakiś tekst.

Filtr godzinowy możesz przedstawić w postaci <select> z opcjami od godziny 11 i iteracją co 30 minut do godziny 18