Aplikacje internetowe 1 (AI1) – laboratorium nr 4

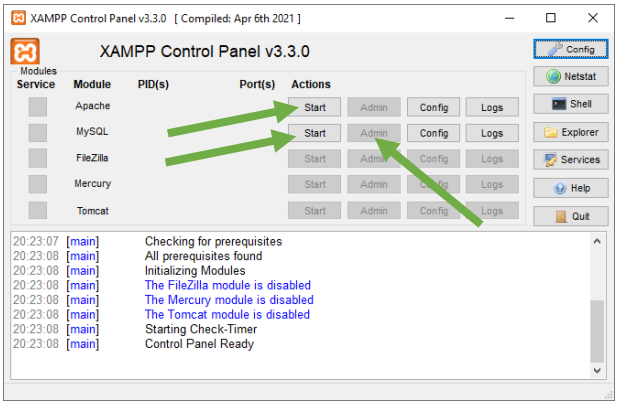
Żądania HTTP, formularze, PDO

Początek laboratorium:

• pobrać na pulpit archiwum `Lab004_AI1_start.zip` w którym umieszczone są pliki potrzebne do wykonania zadań oraz rozpakować to archiwum,

• przejść do rozpakowanego folderu oraz otworzyć folder `lab4` w *VSCode*,

• włączyć panel zarządzania *XAMPP* oraz uruchomić moduły *Apache* oraz *MySQL* (*Start* x2), następnie kliknąć przycisk *Admin* przy *MySQL*, w celu przejścia do panelu *phpMyAdmin*,



Zadania (*HTTP*, *PHP*):

Zadanie 4.1:
Zapoznać się z następującymi zagadnieniami dotyczącymi *protokołu HTTP*:
• *żądanie HTTP* (*request*),
• *części żądania HTTP* (pierwsza linijka, *nagłówki*, *ciało*),
• pierwsza linijka *żądania HTTP* (*metoda GET/POST*, *adres URL*, *wersja protokołu*),
• *nagłówki żądania HTTP* (np. *Host*, *User-Agent*, *Accept*, *Referer*, *Cookie*),
• *ciało żądania* (*request body*),
• odpowiedź *HTTP* (*response*),
• *części odpowiedzi HTTP* (*pierwsza linijka*, *nagłówki*, *ciało*),
• pierwsza linijka *odpowiedzi HTTP* (wersja protokołu, kod statusu),
• *nagłówki odpowiedzi HTTP* (np. *Allow*, *Content-Type*, *Set-Cookie*, *Server*, *Location*),
• *ciało odpowiedzi* (*response body*).
Ponadto, *kodowanie procentowe* (%...).

https://cdn.sekurak.pl/Podstawy_HTTP_v10.pdf

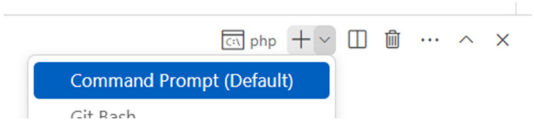
https://developer.mozilla.org/en-US/docs/Glossary/Request_header
https://oxylabs.io/blog/http-headers-explained

https://pl.wikipedia.org/wiki/Kodowanie_procentowe

Zadanie 4.2:

Otworzyć terminal *cmd (Command Prompt)* w *VSCode*.
Uruchomić serwer *php* dla folderu zawierającego pliki zadania.
Otworzyć drugą kartę terminala *cmd (Command Prompt)* w *VSCode*.
W drugiej karcie terminala wykonać komendę *cURL*, w celu wykonania *żądania HTTP*.
W wyświetlonej zawartości wskazać konkretne elementy *żądania* oraz *odpowiedzi HTTP*.

```
php -S localhost:8008
```



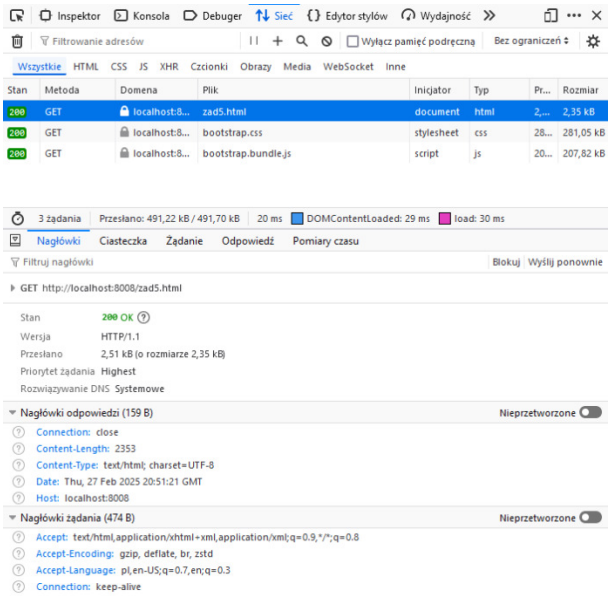

```
curl -v http://localhost:8008/zad5.html
```

```
*   Trying 127.0.0.1:8008...
*   Trying ::1:8008...
* Connected to localhost (::1) port 8008 (#0)
> GET /zad5.html HTTP/1.1
> Host: localhost:8008
> User-Agent: curl/7.83.1
> Accept: */*
>
* Mark bundle as not supporting multiuse
< HTTP/1.1 200 OK
< Host: localhost:8008
< Date: Thu, 27 Feb 2025 20:49:36 GMT
< Connection: close
< Content-Type: text/html; charset=UTF-8
< Content-Length: 2353
<
<!doctype html>
<html lang="pl" data-bs-theme="">
  <head>
    <link rel="icon" href="data:;base64,iVBORw0KGgo=">
    <meta charset="utf-8">
    <meta name="viewport" content="width=device-width, initial-scale=1">
    <title>Zad5</title>
```

Zadanie 4.3:

W przeglądarce internetowej przejść pod wskazany adres oraz:
• otworzyć *Panel dla programistów* (*F12*) i przejść do zakładki *Sieć*,
• odświeżyć stronę (*F5*),
• kliknąć na pierwsze żądanie na liście,
• wskazać analogicznie konkretne elementy *żądania* oraz *odpowiedzi HTTP*. (nieprzetworzone)

W przeglądarce internetowej przejść pod adres: http://localhost:8008/zad5.html



Zadanie 4.4:
W panelu *phpMyAdmin* utworzyć nową bazę **`ai1_lab4`**.
W pliku *zad4.sql* znajduje się skrypt tworzący schemat tabeli **`questions`**, która będzie pozwalała na przechowywanie pytań od zainteresowanych osób. Wykonać go w tej bazie.

http://localhost/phpmyadmin

Zadanie 4.5:
W pliku *zad5.html* znajduje się formularz „*Zapytanie o ofertę*”. Zapoznać się jego polami. Uzupełnić jego kod w niezbędne atrybuty odpowiedzialne za umożliwienie przesyłania danych:
• określić odpowiednią metodę wysyłania formularza (*GET* lub *POST*),
• określić ścieżkę skryptu przetwarzającego dane przesłanego formularza (do pliku *zad6.php* z następnego zadania),
• zdefiniować atrybut *name* dla każdego z pól formularza,
• zdefiniować atrybut *value* dla wartości niektórych pól formularza,
• zdefiniować odpowiedni przycisk do wykonania przesłania formularza.

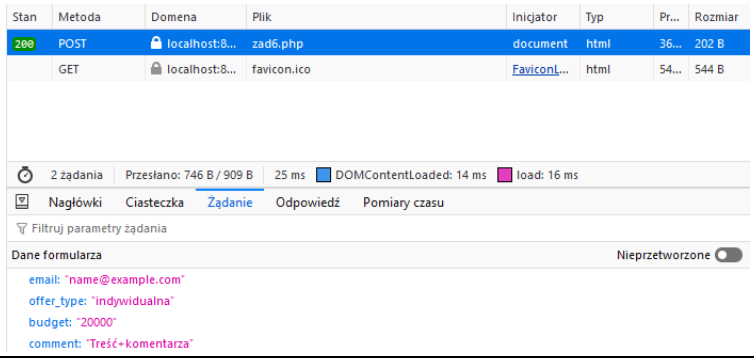
http://localhost:8008/zad5.html
https://www.w3schools.com/tags/ref_httpmethods.asp
https://www.w3schools.com/php/php_forms.asp

Zadanie 4.6:
W pliku *zad6.php* zrealizować odebranie danych z formularza oraz za pomocą połączenia poprzez *PHP Data Objects* (*PDO*) z bazą **`ai1_lab4`** wstawienie nowego rekordu z danymi pytania pochodzącymi z przesyłanego formularza.

https://www.w3schools.com/php/php_superglobals_post.asp
https://phpdelusions.net/pdo_examples/insert

Zadanie 4.7:
Następnie uzupełnić dane formularza oraz:
• otworzyć *Panel dla programistów* (*F12*) i przejść do zakładki *Sieć*,
• kliknąć przycisk przesłania formularza,
• kliknąć zaistniałe żądanie,
• wskazać analogicznie konkretne elementy *żądania* oraz *odpowiedzi HTTP*, w tym w szczególności w zakładce „*Żądanie*” jakie dane zostały były obecne w formularzu (zawartość ciała żądania).
Sprawdzić zawartość tabeli **`questions`**.

http://localhost:8008/zad5.html



Zadanie 4.8:
Przejść do pliku *zad8.php* i wykorzystać *PDO* do pobrania zawartości tabeli **`questions`** oraz wyświetlić dane wszystkich pytań w dynamicznie generowanej tabelce na stronie.

W przeglądarce internetowej przejść pod adres: http://localhost:8008/zad8.php

https://www.php.net/manual/en/pdostatement.fetchall.php

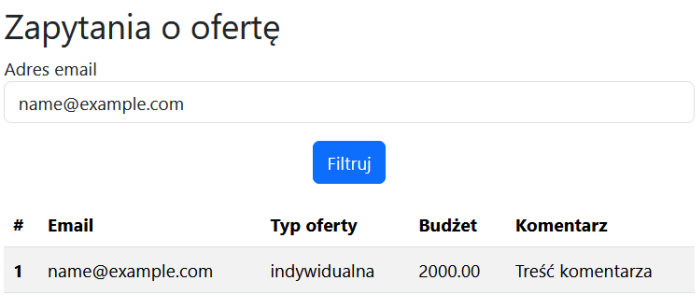
Zadanie 4.9:
Skopiować zawartość pliku *zad8.php* do pliku *zad9.php*.
Nad tabelką dodać (mały) formularz z jednym polem tekstowym **`email`** oraz przyciskiem *Filtruj* (formularz ma używać metody *GET*).
Po kliknięciu *Filtruj* w tabelce mają się wyświetlać tylko dane pytań zadane przez posiadacza danego adresu *email* (wpisanego do pola formularza) – operacja *filtrowania*.
Przekazywanie parametru **`email`** zrealizować w *ciągu zapytania* (ang. *query string'u*/*Query parameters*).
Pole *email* formularza powinno zawierać zawsze ostatnio wyszukiwaną frazę.

W przeglądarce internetowej przejść pod adres: http://localhost:8008/zad9.php

https://www.w3schools.com/php/func_var_isset.asp
https://www.w3schools.com/php/func_var_empty.asp

https://en.wikipedia.org/wiki/Query_string

http://localhost:8008/zad9.php?email=name%40example.com

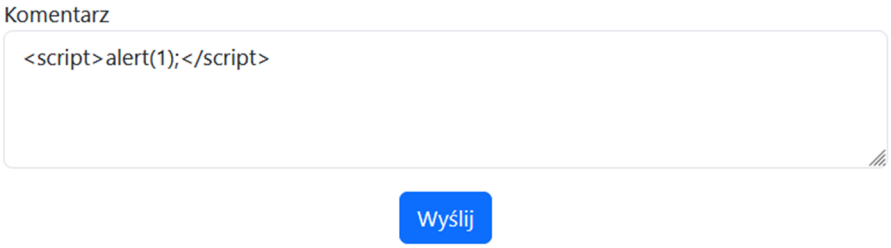


Zadanie 4.10: *
Zaproponować i uzupełnić walidację formularza (*zad5.html*) *HTML5* – po stronie klienta, tak aby próbowała zapobiegać przesyłaniom formularza np.:
• z podanym ujemnym *budżetem*,
• z *budżetem* innym niż liczba,
• z pustym adresem *email*, itp.

https://www.w3schools.com/html/html_form_attributes.asp
https://www.htmlgoodies.com/html5/validations-in-html5-forms

Zadanie 4.11: *
Poprzez formularz (*zad5.html*) dodać nowe pytanie z następującym komentarzem.
Następnie przejść na stronę z tabelą pytań (*zad8.php*).
Jakiej podatności w aplikacjach internetowych dotyczy zaistniałe zdarzenie?
Zmodyfikować plik *zad8.php* w celu zapobiegnięcia jej występowaniu.

W przeglądarce internetowej przejść pod adres: http://localhost:8008/zad5.html



http://localhost:8008/zad8.php

np.
https://www.php.net/manual/en/function.htmlspecialchars.php

Zadanie 4.12: *

Zakładając, że wartości pochodzą od użytkownika (w tych miejscach są przypisane przykładowe wartości), uwzględniając kwestie bezpieczeństwa, realizacja wstawienia nowego rekordu do bazy może zrealizowana na dwa sposoby.
Wyjaśnić, który sposób może spowodować wystąpienie podatności w aplikacji internetowej oraz podać nazwę tej podatności.

propozycja nr 1:

```
# Użytkownik w formularzu uzupełnił wartości poniższych zmiennych
$name = '...'; $surname = '...'; $age = '...';
$statement = $link->prepare(
"INSERT INTO students(name, lastname, age) VALUES ('.$name.','.$surname.','.$age.')"
            );
$statement->execute();
```

propozycja nr 2:

```
$statement = $link->prepare(
"INSERT INTO students(name, lastname, age) VALUES(?,?,?)");
# Użytkownik w formularzu uzupełnił wartości poniższych ciągów znaków
$statement->execute(array("...","...","..."));
```

* – zadania/podpunkty do samodzielnego dokończenia/wykonania,

* – zadania/podpunkty dla zainteresowanych.

Po zakończonym laboratorium należy skasować wszystkie pobrane oraz utworzone przez siebie pliki z komputera w sali laboratoryjnej.