

Technologie internetowe

lista zadań nr 6

Na potrzeby tej listy zadań potrzebne będzie utworzenie prostego skryptu po stronie serwera, przy czym rekomendowane jest użycie PHP i z myślą o tej technologii treści zadań zostały sformułowane. Oczywiście można także wybrać dowolną inną ulubioną technologię, wtedy jednak trzeba samodzielnie dostosować treści zadań. Do uruchomienia skryptu można wykorzystać Microsoft Azure.

Przykładowy skrypt w PHP (*test.php*), który odsyła otrzymaną zmienną *test*:

```
<?php echo $_REQUEST["test"]; ?>
```

1. Korzystając z programu *nslookup* (lub innego analogicznego) ustal adresy IP serwerów WWW dla adresów `www.ii.uni.wroc.pl` i `www.uj.edu.pl`. Sprawdź również, jakie serwery poczty obsługują obie domeny (ponownie korzystając z programu *nslookup* lub analogicznego).

[1p]

2. Odnajdź plik `hosts` w swoim systemie operacyjnym i dodaj w nim wiersz:

```
79.96.11.200 pawel.ii.uni.wroc.pl
```

Następnie otwórz okno przeglądarki w trybie *prywatnym/incognito* i wprowadź adres `http://pawel.ii.uni.wroc.pl/`. Opisz i wyjaśnij, co się stało. Teraz wymień powyższą linią na:

```
156.17.4.11 pawel.ii.uni.wroc.pl
```

Zastanów się nad przyczyną tego się stało i spróbuj to wyjaśnić. Na końcu usuń dodaną linię z pliku `hosts`.

[1p]

3. W programie Postman pobierz stronę `http://itcourses.eu`. Następnie uruchom polecenie `netstat` i sprawdź z jakiego portu źródłowego połączenie zostało nawiązane. Aby przyspieszyć prezentację listy możesz pominąć rozwiązywanie nazw używając opcji `-n`. Powtórz całość jeszcze raz, ale tym razem zmierz jak długo (mniej więcej) połączenie jest otwarte. Na koniec używając narzędzia dostępnego pod adresem `https://tools.keycdn.com/geo` sprawdź lokalizację serwera. Sprawdź też lokalizację serwera, na którym hostujesz rozwiązania w chmurze (np. Azure czy Amazon). Czy zgadza się region, który został wybrany przy tworzeniu serwisu?

[1p]

4. Wyjaśnij na czym polega „bezstanowość” protokołu HTTP. W narzędziach deweloperskich przeglądarki Google Chrome (F12), w zakładce `network` obejrzyj proces pobierania wybranej, „cięższej” strony (np. `http://www.onet.pl/`).

[1p]

5. Za pomocą programu **Fiddler** lub **Postman** pobierz poprzez protokół HTTP i metodę GET stronę główną Google (<http://www.google.com/>). Następnie dodaj do żądania nagłówek **Accept-Language**. Sprawdź, jakie wyniki zostaną zwrócone po wybraniu języków polskiego, niemieckiego i francuskiego.
[1p]
6. Napisz kod w PHP, który wyświetli zmienne wysłane metodą POST (tablica `$_POST`, patrz przykład na początku listy) i opublikuj w portalu Microsoft Azure. Za pomocą programu **Fiddler** lub **Postman** wyślij do tego skryptu kilka par klucz/wartość. Zobacz jakie znaczenie ma parametr **Content-Length**. Dodaj zapytanie do zasobu, tak żeby jednocześnie wypełniona była tablica `$_GET`. Jakie wartości zawiera tablica `$_REQUEST` w przypadku, zmienne w tablicach `$_GET` i `$_POST` mają taką samą nazwę?
[1p]
7. Poprzednie zadanie przerób tak, żeby dane były wysyłane z kodowaniem **multipart/form-data**. Następnie przygotuj przykład, który przy użyciu tego kodowania wyśle plik binarny (pamiętaj, że cały czas naszym klientem jest program **Fiddler** lub **Postman**). Obejrzyj w Fiddlerze ruch i zobacz, jak wygląda wysłane żądanie.
[1p]
8. Korzystając z programu **Postman** wyślij kilka pól do skryptu PHP, który odsyła je z powrotem (patrz przykład na początku listy). Użyj programu **Fiddler** do podejrzenia żądań i odpowiedzi. Postaraj się zrozumieć znaczenie poszczególnych pól. Używając mechanizmu „automatic breakpoints” podmień wysłane dane. Na koniec zobacz konfigurację serwerów proxy przeglądarki IE przed i po uruchomieniu programu **Fiddler**.
[1p]
9. Korzystając z programu **Postman** wyślij żądania po treści różnego typu: plik HTML, obrazek JPG, obrazek PNG, plik XML (<https://www.w3schools.com/xml/note.xml>) i plik JSON (<https://tools.learningcontainer.com/sample-json.json>). Obejrzyj za każdym razem wartość pola *Content-Type*. Następnie, korzystając z programu **Postman** wyślij żądanie do jakiegoś pliku PNG, ale w odpowiedzi za pomocą **Fiddlera** podmień pole *Content-Type* na *text/html*. Czy **Postman** pokazał obrazek? Wyciągnij wnioski.
[1p]
10. Korzystając z funkcji wyświetlającej bieżący czas, przygotuj prosty przykład, za pomocą którego zademonstrujesz mechanizm buforowania odpowiedzi w przeglądarce. Jakimi nagłówkami można sterować zachowanie przeglądarki?
[1p]

Paweł Rajba