

Technologie internetowe

lista zadań nr 7

Na potrzeby tej listy zadań potrzebne będzie utworzenie prostego skryptu po stronie serwera. Rekomendowane jest użycie PHP i w kontekście tej technologii zostały sformułowane treści poniższych zadań. Do uruchomienia PHP można zainstalować własne środowisko, można też wykorzystać rozwiązania w chmurze, np. założyć konto na <http://c9.io/> i utworzyć *workspace* dla PHP. Oczywiście można także wybrać dowolną inną ulubioną technologię, wtedy jednak trzeba samodzielnie dostosować treść zadań do tej technologii.

1. Wyjaśnij na czym polega “bezstanowość” protokołu HTTP i jakie w tym kontekście mają znaczenie sesje. W narzędziach deweloperskich przeglądarki Google Chrome (F12), w zakładce network obejrzyj proces pobierania wybranej, „cięższej” strony (np. <http://www.onet.pl/>). 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]
2. Napisz kod w PHP, który wyświetli zmienne wysłane metodą POST (tablica `$_POST`). Za pomocą programu Fiddler lub Postman wyślij do tego skryptu kilka par klucz/wartość. Zobacz jakie znaczenie ma parametr **Content-Length**. Dodaj zapytanie do zasobu, tak żeby jednocześnie wypełniona była tablica `$_GET`. Jakie wartości zawiera tablica `$_REQUEST` w przypadku, zmienne w tablicach `$_GET` i `$_POST` mają taką samą nazwę? Na koniec dodaj fragment kodu, który wyświetli zawartość zmiennej `$_SERVER`. Przeglądaj jej zawartość.
[1p]
3. 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]
4. Utwórz w HTML prosty formularz, który wysyła dane do skryptu PHP. Wyświetl go w przeglądarce. Użyj programu Fiddler do podejrzenia żądań i odpowiedzi. Postaraj się zrozumieć znaczenie poszczególnych pól. Używając mechanizmu “automatic breakpoints” podmień dane wysłane z formularza. Na koniec zobacz konfigurację serwerów proxy przeglądarki IE przed i po uruchomieniu programu Fiddler.
[1p]
5. Przygotuj prosty przykład, za pomocą którego zademonstrujesz mechanizm buforowania odpowiedzi w przeglądarce. Jakimi nagłówkami można sterować zachowanie przeglądarki?
[1p]

6. Zapoznaj się zagadnieniem żądań warunkowych:
https://developer.mozilla.org/en-US/docs/Web/HTTP/Conditional_requests.
Przygotuj przykład(y) z użyciem co najmniej dwóch nagłówków typu `If-*`.
[1p]
7. Wyjaśnij jak działa mechanizm ciastek. Obejrzyj ruch w Fiddlerze i wskaż pozycje, które dotyczą ciastek. Napisz prosty skrypt do wyświetlania ciastek i wyślij do niego ciastka z Fiddlera. Znajdź też ciastka w narzędziach deweloperskich swojej ulubionej przeglądarki.
[1p]
8. Napisz prosty skrypt w JavaScript, który wyświetla zawartość ciastek. Wyjaśnij znaczenie flagi `HttpOnly` w ciastkach i sprawdź, jak jej ustawienie wpływa do pobierania wartości z ciastek w utworzonym skrypcie.
[1p]
9. Sprawdź czym się różni ciastko „persistent cookie” od ciastka per sesja. Ustal jak przechowywane są ciastka w co najmniej dwóch przeglądarkach. Czy jest różnica w składowaniu ciastek „trwałych” i sesyjnych?
[1p]
10. Każde ciastko może mieć ustawioną domenę i ścieżkę. Wyjaśnij ich znaczenie i zrób przykłady, które to znaczenie pokażą.
[1p]

Paweł Rajba