ARKUSZ ĆWICZENIOWY • KWALIFIKACJA INF.03

Portal biblioteki

Rejestracja, logowanie i sesje użytkownika oraz operacje na bazie danych

Baza:

```
CREATE TABLE uzytkownicy (
   id    INT AUTO_INCREMENT PRIMARY KEY,
   login VARCHAR(50) NOT NULL UNIQUE,
   haslo VARCHAR(40) NOT NULL        -- sha1() zwraca zawsze 40 znaków
);

-- konto testowe: login = jan, hasło = haslo123 (zapisane jako sha1)
INSERT INTO uzytkownicy (login, haslo)
VALUES ('jan', '04c72343945e2a6ef09221862164ac3a9e914373');
```

Zadanie

Wykonaj aplikację internetową portalu biblioteki, w której użytkownik może założyć konto, zalogować się i wylogować, a strona panelu jest dostępna tylko dla osoby zalogowanej. Wykonaj także operacje administracyjne na bazie danych. Do wykonania zadania wykorzystaj plik `baza.sql`.

Aplikacja składa się z plików: `rejestracja.php`, `logowanie.php`, `panel.php`, `wyloguj.php` oraz arkusza stylów `styl.css`. Strony należy wykonać z zachowaniem zasad poprawnego HTML5. Kod skryptów powinien być czytelny, z wcięciami i komentarzami.

Operacje na bazie danych

Za pomocą narzędzia phpMyAdmin zaimportuj plik `baza.sql`. Skrypt tworzy bazę `biblioteka` oraz tabelę `uzytkownicy` z jednym kontem testowym.

Struktura tabeli uzytkownicy

Pole	Typ	Opis
`id`	INT, klucz główny, AUTO_INCREMENT	identyfikator użytkownika
`login`	VARCHAR(50), UNIQUE, NOT NULL	login użytkownika
`haslo`	VARCHAR(40), NOT NULL	hasło zaszyfrowane funkcją `sha1()`

Utwórz następujące kwerendy (zapytania SQL) i wykonaj je w bazie `biblioteka`:

1. Kwerenda tworząca na serwerze bazy danych nowego użytkownika o nazwie **uczen** z hasłem **zse**, który może łączyć się wyłącznie z hosta `localhost`.

2. Kwerenda nadająca użytkownikowi **uczen** uprawnienie do wybierania danych (**SELECT**) w tabeli `uzytkownicy`, ograniczone **wyłącznie do kolumny** `login`. Użytkownik nie może odczytywać pozostałych kolumn ani wykonywać innych operacji.
3. Kwerenda tworząca tabelę **historia_zmian** o polach: `id_zmiany` — klucz główny z autonumeracją, `stary_login` i `nowy_login` — tekstowe o długości do 50 znaków, `data_zmiany` — data i godzina zmiany.
4. Kwerenda tworząca **wyzwalacz** (trigger) o nazwie `zmiana_loginu`, który po każdej modyfikacji rekordu w tabeli `uzytkownicy` automatycznie dopisuje do tabeli `historia_zmian` dotychczasowy login, nowy login oraz bieżącą datę i godzinę.
5. Kwerenda usuwająca **całą zawartość** tabeli `historia_zmian` (bez usuwania samej tabeli).

> **Zapis wyników**
>
> Treść wszystkich kwerend zapisz w pliku `kwerendy.txt`, każdą w osobnej linii lub bloku, z numerem kwerendy w komentarzu.
>
> Wykonaj zrzuty ekranu potwierdzające wykonanie kwerend i zapisz je jako `kw1.png` … `kw5.png`. Zrzuty mają obejmować treść kwerendy i komunikat lub wynik jej wykonania.
>
> Dla kwerendy 4 przed wykonaniem zrzutu `kw4.png` zmień w tabeli `uzytkownicy` login dowolnego użytkownika i pokaż na zrzucie zawartość tabeli `historia_zmian`.
>
> Dla kwerendy 2 wykonaj dodatkowy zrzut `kw2_test.png`: zaloguj się do phpMyAdmin jako **uczen** i pokaż, że widzi on tylko kolumnę `login`.

Witryna internetowa

Wszystkie strony mają język polski, kodowanie UTF-8 i są połączone z arkuszem stylów `styl.css`. Tytuły stron (widoczne na karcie przeglądarki) podano przy opisie każdej strony.

Strona rejestracja.php

- Tytuł strony: **Rejestracja**
- Nagłówek drugiego stopnia o treści: „Załóż konto”.
- Formularz wysyłany metodą **POST** do pliku `rejestracja.php`, zawierający kolejno: etykietę „Login:” i pole tekstowe `login`, etykietę „Hasło:” i pole hasła `haslo`, etykietę „Powtórz hasło:” i pole hasła `haslo2`, przycisk „Zarejestruj”.
- Pod formularzem akapit z odnośnikiem „Masz już konto? Zaloguj się” prowadzącym do `logowanie.php`.
- Pod odnośnikiem wyświetla się komunikat wygenerowany przez skrypt.

Strona logowanie.php

- Tytuł strony: **Logowanie**
- Nagłówek drugiego stopnia o treści: „Logowanie”.
- Formularz wysyłany metodą **POST** do pliku `logowanie.php`: etykieta „Login:” i pole tekstowe `login`, etykieta „Hasło:” i pole hasła `haslo`, przycisk „Zaloguj”.
- Pod formularzem akapit na komunikat skryptu, a pod nim akapit z odnośnikiem „Nie masz konta? Zarejestruj się” prowadzącym do `rejestracja.php`.

Strona panel.php

- Tytuł strony: **Panel użytkownika**
- Nagłówek drugiego stopnia: „Panel użytkownika”.
- Akapit z powitaniem: „Witaj, **login**!”, gdzie login jest pogrubiony i pobrany z sesji.
- Akapit: „Tę stronę widzą tylko zalogowani użytkownicy.”
- Akapit z odnośnikiem „Wyloguj” prowadzącym do `wyloguj.php`.

Styl CSS witryny

Plik `styl.css` zawiera formatowanie:

Selektor	Wymagane formatowanie
dla całej strony	krój czcionki Arial, kolor tła #EEEEEE, wyrównanie tekstu do środka
formularza	kolor tła biały, szerokość 300 px, wyśrodkowany na stronie (marginesy automatyczne), marginesy wewnętrzne 20 px, obramowanie 1 px ciągłe w kolorze #CCCCCC
pól input	szerokość 90%, marginesy wewnętrzne 5 px, margines górny 5 px i dolny 15 px
przycisku	kolor tła #2A6EBB, kolor czcionki biały, marginesy wewnętrzne 8 px (góra, dół) i 20 px (lewo, prawo), brak obramowania, kursor w kształcie dłoni
odnośników	kolor czcionki #2A6EBB

Skrypty PHP

Połączenie z serwerem bazy danych: host `localhost`, użytkownik `root`, hasło puste, baza `biblioteka`. Skrypty wykonaj z użyciem rozszerzenia **mysqli**. Po zakończeniu operacji na bazie połączenie musi zostać zamknięte.

Skrypt w pliku rejestracja.php

1. Działa po wysłaniu formularza (sprawdza, czy istnieje pole `login` przesłane metodą POST).
2. Jeżeli którekolwiek z trzech pól jest puste, wyświetla komunikat: „Wypełnij wszystkie pola!”.
3. W przeciwnym razie, jeżeli hasła są różne, wyświetla: „Hasła nie są identyczne!”.
4. W przeciwnym razie łączy się z bazą i sprawdza, czy w tabeli `uzytkownicy` istnieje już podany login. Jeżeli tak, wyświetla: „Taki login już istnieje.”.
5. Jeżeli login jest wolny, szyfruje hasło funkcją `sha1()`, dodaje nowy rekord do tabeli `uzytkownicy` i wyświetla: „Konto zostało założone.”.

Skrypt w pliku logowanie.php

1. Rozpoczyna lub wznawia sesję. Kod skryptu znajduje się **przed** kodem HTML strony.
2. Działa, gdy przesłano oba pola: `login` i `haslo`.
3. Szyfruje podane hasło funkcją `sha1()` i wyszukuje w tabeli `uzytkownicy` rekord z podanym loginem i hasłem.
4. Jeżeli znaleziono dokładnie jeden rekord: zapisuje login w zmiennej sesyjnej o kluczu `uzytkownik` i przekierowuje na stronę `panel.php`.
5. W przeciwnym razie w miejscu komunikatu wyświetla: „Błędny login lub hasło.”.

Skrypt w pliku panel.php

1. Wznawia sesję.
2. Jeżeli zmienna sesyjna `uzytkownik` nie istnieje, przekierowuje na stronę `logowanie.php` i kończy działanie skryptu.
3. W przeciwnym razie strona wyświetla powitanie z loginem pobranym z sesji.

Skrypt w pliku wyloguj.php

1. Wznawia sesję.
2. Jeżeli istnieje zmienna sesyjna `uzytkownik`, usuwa ją i niszczy sesję.
3. Przekierowuje na stronę `logowanie.php` i kończy działanie skryptu.

> **Sprawdzenie działania**
>
> Zarejestruj nowe konto, zaloguj się na nie, a następnie wyloguj. Po wylogowaniu wpisz w pasku adresu `panel.php` — strona powinna przekierować do logowania.
>
> Wykonaj zrzut ekranu strony `panel.php` po zalogowaniu i zapisz go jako `panel.png`.

Pliki do oddania

W folderze `nazwisko_imie` muszą się znaleźć:

Plik	Zawartość
`kwerendy.txt`	treść kwerend 1–5
`kw1.png` … `kw5.png`, `kw2_test.png`	zrzuty ekranu z wykonania kwerend
`rejestracja.php`, `logowanie.php`, `panel.php`, `wyloguj.php`	strony i skrypty
`styl.css`	arkusz stylów
`panel.png`	zrzut ekranu panelu po zalogowaniu

Konto testowe z pliku `baza.sql`: login **jan**, hasło **haslo123**.



