

### Zadanie egzaminacyjne

Wykonaj aplikację internetową portalu prognozy pogody, wykorzystując pakiet XAMPP oraz edytor z zaznaczający składnię.

Aby wykonać zadanie, zaloguj się na konto **Egzamin** bez hasła. Na pulpicie znajdziesz archiwum z plikami o nazwie *pliki2.zip* zabezpieczone hasłem: **3MeTeo#**

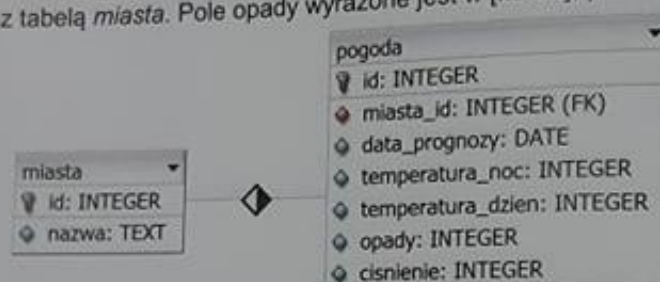
Archiwum należy rozpakować.

Należy utworzyć folder o nazwie odpowiadającej numerowi PESEL zdającego lub w przypadku jego braku innemu numerowi, którym został podpisany arkusz. Archiwum należy rozpakować do tego folderu.

**Ważne!** Wszystkie wykonane w czasie egzaminu pliki w tym zrzuty ekranu, pliki strony, stylów należy umieścić w tym folderze.

### Operacje na bazie danych

Do wykonania zadania należy użyć tabel: *miasta* i *pogoda* przedstawionych na Obrazie 1. Tabela *pogoda* jest powiązana relacją z tabelą *miasta*. Pole opady wyrażone jest w [mm/h], pole ciśnienie w [hPa]



Obraz 1. Tabele miasta i pogoda

Uruchom usługi MySQL i Apache za pomocą XAMPP Control Panel. Za pomocą narzędzia phpMyAdmin wykonaj podane operacje na bazie danych:

- Utwórz bazę danych o nazwie *prognoza*
- Do bazy *prognoza* zaimportuj tabele z pliku *baza2.sql* z rozpakowanego archiwum
- Wykonaj zrzut ekranu po imporcie. Zrzut zapisz w folderze z numerem PESEL lub w przypadku jego braku w folderze z innym numerem, którym został podpisany arkusz, w formacie PNG i nazwij *import*. Nie kadruj zrzutu. Powinien on obejmować cały ekran monitora, z widocznym paskiem zadań. Na zrzucie powinny być widoczne elementy wskazujące na poprawnie wykonany import tabel
- Zapisz i wykonaj podane zapytania SQL działające na bazie *prognoza*. Zapytania zapisz w pliku *kwerendy.txt*, w folderze z numerem PESEL lub w przypadku jego braku w folderze z innym numerem, którym został podpisany arkusz. Wykonaj zrzuty ekranu przedstawiające wyniki działania kwerend. Zrzuty zapisz w formacie JPEG i nadaj im nazwy *kw1*, *kw2*, *kw3*, *kw4*. Zrzuty powinny obejmować cały ekran monitora z widocznym paskiem zadań
- Zapytanie 1: wybierające wszystkie pola z tabeli *pogoda* dla miast o id równym 1, posortowane rosnąco według daty prognozy
- Zapytanie 2: wybierające jedynie pola id oraz *data\_prognozy* z tabeli *pogoda* dla tych rekordów, dla których opady przekraczają 30 mm/h
- Zapytanie 3: wybierające jedynie pola *data\_prognozy*, *temperatura\_noc* z tabeli *pogoda* oraz *nazwa* z tabeli *miasta* dla temperatur nocnych poniżej 6 stopni. Zapytanie wykorzystuje relację
- Zapytanie 4: dodające do tabeli *miasta* kolumnę wojewodztwo dowolnego typu tekstowego

DATA	TEMPERATURA W NOCY	TEMPERATURA W DZIEŃ	OPADY [mm/h]	CISNIENIE [hPa]
2019-05-10	15	23		
2019-05-11	14	22	0	1020
2019-05-12	10	14	0	1020
2019-05-13	6	15	0	1020
2019-05-14	5	11	0	1020
2019-05-15	6	11	7	1000
2019-05-16	6	15	33	1000
2019-05-17	12	20	32	997
2019-05-31	15	23	11	997
			33	1020



Stronę wykonał: 000000000000

Obraz 2. Witryna internetowa

#### Przygotowanie grafiki:

- Plik *logo.png*, wypakowany z archiwum, należy przeskalować z zachowaniem proporcji tak, aby jego wysokość wynosiła dokładnie 80 px; plik ma zachować przezroczystość
- Plik *obraz.jpg*, wypakowany z archiwum, należy przeskalować z zachowaniem proporcji tak, aby jego wysokość wynosiła dokładnie 200 px

#### Cechy witryny:

- Składa się ze strony o nazwie *pogoda.php*
- Zastosowany właściwy standard kodowania polskich znaków
- Tytuł strony widoczny na karcie przeglądarki: „Prognoza pogody Wrocław”
- Arkusz stylów w pliku o nazwie *styl2.css* prawidłowo połączony z kodem strony
- Podział strony na bloki: na górze trzy bloki banera: lewy, środkowy, prawy, poniżej blok główny, poniżej dwa bloki: lewy i prawy, na dole blok stopki. Podział zrealizowany za pomocą znaczników sekcji tak, aby po uruchomieniu w przeglądarce wygląd układu bloków był zgodny z Obrazem 2
- Zawartość lewego bloku banera: obraz *logo.png* z tekstem alternatywnym o treści „meteo”
- Zawartość środkowego bloku banera: nagłówek pierwszego stopnia o treści „Prognoza dla Wrocławia”
- Zawartość prawego bloku banera: akapit (paragraf) o treści „maj, 2019 r.”
- Zawartość bloku głównego:
  - Tabela o pięciu kolumnach
  - Wiersz nagłówkowy tabeli zawiera wpisy: „DATA”, „TEMPERATURA W NOCY”, „TEMPERATURA W DZIEŃ”, „OPADY [mm/h]”, „CIŚNIENIE [hPa]”
  - Wiersze z danymi tabeli są wypełnione za pomocą skryptu
- Zawartość bloku lewego: *obraz.jpg* z tekstem alternatywnym o treści „Polska, Wrocław”
- Zawartość bloku prawego: odnośnik do pliku *kwerendy.txt* z treścią: „Pobierz kwerendy”
- Zawartość bloku stopki: akapit (paragraf) o treści: „Stronę wykonał: ”, dalej wstawiony numer PESEL zdającego lub w przypadku jego braku inny numer, którym został podpisany arkusz



### Styl CSS witryny internetowej

Cechy formatowania CSS:

- Są zawarte wyłącznie w zewnętrznym arkuszu stylów
- Domyślne dla całej strony: krój czcionki Helvetica
- Wspólne dla lewego i prawego bloku banera: kolor tła RGB 70, 130, 180; biały kolor czcionki, szerokość 25%, wysokość 80 px
- Dla środkowego bloku banera: kolor tła RGB 70, 130, 180; biały kolor czcionki, szerokość 50%, wysokość 80 px, wyrównanie tekstu do środka
- Dla bloku głównego: kolor tła RGB 224, 255, 255; marginesy wewnętrzne 40 px
- Dla bloku lewego i bloku prawego: kolor tła RGB 70, 130, 180; szerokość 50%, wysokość 200 px, wyrównanie tekstu do środka
- Dla bloku stopki: kolor tła RGB 224, 255, 255
- Dla znacznika tabeli: obramowanie 1 px, linią ciągłą o niebieskim kolorze, szerokość 100%, obramowanie połączone za pomocą jednej linii, jak na Obrazie 2
- Dla znacznika komórki danych: obramowanie 1 px, linią ciągłą o niebieskim kolorze
- Dla znacznika komórki nagłówka tabeli: obramowanie 1 px, linią ciągłą o niebieskim kolorze, kolor tła niebieski, kolor czcionki biały

### Skrypt połączenia z bazą

W tabeli 1 podano wybór funkcji PHP do obsługi bazy danych. Wymagania dotyczące skryptu:

- Napisany w języku PHP
- Skrypt łączy się z serwerem bazodanowym na *localhost*, użytkownik **root** bez hasła, baza danych o nazwie *progniza*
- Skrypt wysyła do bazy danych zapytanie 1
- Każdy zwrócony zapytaniem wiersz jest wyświetlany w osobnym wierszu tabeli w bloku głównym. Dane pobrane z bazy należy umieścić w odpowiednich kolumnach tabeli zgodnie z Obrazem 2
- Na końcu działania skrypt zamyka połączenie z serwerem.

Tabela 1. Wybór funkcji języka PHP do obsługi bazy MySQL i MariaDB

Funkcje biblioteki MySQLi	Zwracana wartość
<code>mysqli_connect(serwer, użytkownik, hasło, nazwa_bazy)</code>	id połączenia lub FALSE, gdy niepowodzenie
<code>mysqli_select_db(id_polaczenia, nazwa_bazy)</code>	TRUE/FALSE w zależności od stanu operacji
<code>mysqli_error(id_polaczenia)</code>	Tekst komunikatu błędu
<code>mysqli_close(id_polaczenia)</code>	TRUE/FALSE w zależności od stanu operacji
<code>mysqli_query(id_polaczenia, zapytanie)</code>	Wynik zapytania
<code>mysqli_fetch_row(wynik_zapytania)</code>	Tablica numeryczna odpowiadająca wierszowi zapytania
<code>mysqli_fetch_array(wynik_zapytania)</code>	Tablica zawierająca kolejny wiersz z podanych w wyniku zapytania lub FALSE, jeżeli nie ma więcej wierszy w wyniku zapytania