**Kwalifikacja INF3 - STYCZEŃ 2024**

**PYTANIE NR 2.**

Kodowanie polskich znaków można zdefiniować w języku HTML za pomocą

|  |
| --- |
|  |
| A. | atrybutu znacznika <p> |
| B. | znacznika <charset> |
| C. | atrybutu znacznika <meta> |
| D. | znacznika <title> |

***Wyjaśnienie poprawnej odpowiedzi:***  
Kodowanie polskich znaków w dokumencie HTML określa się za pomocą atrybutu znacznika <meta>, konkretnie poprzez użycie atrybutu charset w tym znaczniku. Przykładowo, aby zdefiniować kodowanie UTF-8, które wspiera polskie znaki, używa się znacznika <meta charset="UTF-8">. Pozostałe odpowiedzi są niepoprawne: nie istnieje znacznik <charset>, atrybut znacznika <p> nie służy do definiowania kodowania znaków, podobnie jak znacznik <title>, który jest używany do określenia tytułu dokumentu.

**PYTANIE NR 3.**

Przedstawione formatowanie CSS, przy założeniu, że żadne inne formatowanie nie jest zdefiniowane, sprawi, że

[](https://www.testy.egzaminzawodowy.info/question.php?question_id=119139)

|  |
| --- |
|  |
| A. | margines wewnętrzny komórki z napisem Anna będzie miał 10 px, a z napisem Ewa – 30 px |
| B. | marginesy wewnętrzne wszystkich komórek będą wynosiły 10 px |
| C. | margines wewnętrzny komórki z napisem Anna będzie miał 30 px, a z napisem Ewa – 10 px |
| D. | marginesy wewnętrzne wszystkich komórek będą wynosiły 30 px |

***Wyjaśnienie poprawnej odpowiedzi:***  
W kodzie CSS określono, że wszystkie komórki () mają mieć margines wewnętrzny (padding) równy 30px. Następnie dla komórki z napisem "Anna" zdefiniowano styl inline, który nadpisuje poprzednią deklarację i ustawia padding na 10px. Styl inline ma wyższy priorytet niż styl zewnętrzny, dlatego komórka z napisem "Anna" będzie miała padding 10px, a komórka z napisem "Ewa" (dla której nie zdefiniowano stylu inline) odziedziczy styl zewnętrzny, czyli będzie miała padding 30px.

**PYTANIE NR 4.**

Przedstawiono efekt formatowania CSS oraz kod HTML. Jak należy zdefiniować styl, aby osiągnąć takie formatowanie?

[](https://www.testy.egzaminzawodowy.info/question.php?question_id=119138)

|  |
| --- |
|  |
| A. | p.first-letter { font-size: 400%; color: blue; } |
| B. | .first-letter { font-size: 400%; color: blue; } |
| C. | #first-letter { font-size: 400%; color: blue; } |
| D. | p::first-letter { font-size: 400%; color: blue; } |

***Wyjaśnienie poprawnej odpowiedzi:***  
Aby osiągnąć efekt formatowania widoczny na zdjęciu, gdzie pierwsza litera akapitu jest powiększona i niebieska, należy użyć selektora pseudoelementu ::first-letter. Właściwość font-size: 400% powiększa pierwszą literę do czterokrotności normalnego rozmiaru tekstu, a color: blue nadaje jej niebieski kolor. Selektor p::first-letter odnosi się do pierwszej litery każdego akapitu (<p>). Jest to poprawne rozwiązanie, ponieważ pozwala na automatyczne formatowanie pierwszej litery każdego akapitu bez konieczności dodawania dodatkowych klas lub identyfikatorów do kodu HTML.

**PYTANIE NR 5.**

Aby zbadać rozkład ilościowy poszczególnych kolorów zdjęcia, należy użyć

|  |
| --- |
|  |
| A. | rozmycia Gaussa. |
| B. | desaturacji. |
| C. | balansu kolorów. |
| D. | histogramu. |

***Wyjaśnienie poprawnej odpowiedzi:***  
Histogram jest narzędziem używanym w obróbce obrazów do przedstawienia rozkładu ilościowego poszczególnych kolorów na zdjęciu. Pozwala on na analizę intensywności poszczególnych kolorów oraz ich rozkładu na całym obrazie. Desaturacja służy do usuwania nasycenia kolorów, balans kolorów do korekty kolorystycznej, a rozmycie Gaussa do wygładzania obrazu. Tylko histogram pozwala zbadać rozkład ilościowy kolorów.

**PYTANIE NR 6.**

W języku JavaScript wynik działania instrukcji

zmienna1 -= 1;

będzie taki sam, jak instrukcji

|  |
| --- |
|  |
| A. | zmienna1++; |
| B. | zmienna1--; |
| C. | zmienna1 = zmienna1 - 0; |
| D. | zmienna1 === zmienna1 - 1; |

***Wyjaśnienie poprawnej odpowiedzi:***  
Instrukcja

zmienna1 -= 1;

oznacza, że od wartości zmiennej *zmienna1* zostanie odjęta wartość 1. Jest to równoważne zapisowi

zmienna1 = zmienna1 - 1;

. Operacja

zmienna1--;

również dekrementuje wartość zmiennej *zmienna1* o 1, co czyni ją równoważną z instrukcją

zmienna1 -= 1;

. Pozostałe opcje nie odpowiadają tej samej operacji:

zmienna1++;

inkrementuje wartość zmiennej,

zmienna1 = zmienna1 - 0;

nie zmienia wartości zmiennej, a

zmienna1 === zmienna1 - 1;

jest operacją porównania, która zwraca wartość logiczną, a nie operacją arytmetyczną.

**PYTANIE NR 7.**

Aby stworzyć różnicową kopię bazy danych na serwerze MSSQL, należy zastosować klauzulę

|  |
| --- |
|  |
| A. | WITH FORMAT |
| B. | RESTORE |
| C. | DIFFERENTIAL |
| D. | FULL |

***Wyjaśnienie poprawnej odpowiedzi:***  
Aby stworzyć różnicową kopię bazy danych na serwerze MSSQL, należy zastosować klauzulę "DIFFERENTIAL". Pozwala to na utworzenie kopii zawierającej tylko te dane, które zmieniły się od czasu ostatniej pełnej kopii zapasowej. Pozostałe odpowiedzi, takie jak "RESTORE", "WITH FORMAT", "FULL", odnoszą się do innych operacji związanych z bazami danych, takich jak przywracanie bazy danych, formatowanie nośnika kopii zapasowej lub tworzenie pełnej kopii zapasowej, a nie do tworzenia różnicowej kopii zapasowej.

**PYTANIE NR 8.**

O zmiennej predefiniowanej

$\_POST

z języka PHP można powiedzieć, że

|  |
| --- |
|  |
| A. | jest rozszerzoną wersją tablicy  $\_SESSION  . |
| B. | zawiera dane przesłane do skryptu z formularza. |
| C. | jest kopią tablicy  $\_COOKIE  . |
| D. | zawiera dane bezpośrednio przesłane do skryptu z ciasteczka. |

***Wyjaśnienie poprawnej odpowiedzi:***  
Zmienna predefiniowana

$\_POST

w języku PHP służy do przechowywania danych przesłanych do skryptu za pomocą metody POST w formularzu HTML. Jest to globalna tablica asocjacyjna, która zawiera pary klucz-wartość, gdzie klucze to nazwy pól formularza, a wartości to dane wprowadzone przez użytkownika. Pozostałe odpowiedzi są nieprawidłowe, ponieważ

$\_SESSION

służy do przechowywania danych sesji,

$\_COOKIE

przechowuje dane z ciasteczek, a żadna z tych tablic nie jest bezpośrednio związana z metodą POST ani nie jest jej rozszerzeniem.

**PYTANIE NR 9.**

Aby prawidłowo utworzyć relację typu m…n nienarażoną na redundancję danych, należy

|  |
| --- |
|  |
| A. | stworzyć tabelę pomocniczą. |
| B. | posortować przynajmniej jedną z tabel. |
| C. | połączyć bezpośrednio klucze podstawowe obu tabel. |
| D. | połączyć bezpośrednio klucze obce obu tabel. |

***Wyjaśnienie poprawnej odpowiedzi:***  
Aby prawidłowo utworzyć relację typu m…n nienarażoną na redundancję danych, należy stworzyć tabelę pomocniczą, która będzie przechowywać klucze obce obu tabel uczestniczących w relacji. Taka tabela pomocnicza pozwala na efektywne zarządzanie relacjami many-to-many (m…n), eliminując problem redundancji danych i zapewniając integralność danych w bazie. Pozostałe odpowiedzi nie są poprawne, ponieważ sortowanie tabel nie wpływa na strukturę relacji, a bezpośrednie połączenie kluczy obcych lub podstawowych obu tabel nie rozwiązuje problemu redundancji danych w relacji m…n.

**PYTANIE NR 10.**

Baza danych zawiera tabelę *ksiazki* o polach: *tytul*, *id\_autora*, *data\_wypoz*, *id\_czytelnika*. Każdego dnia generowany jest raport książek wypożyczonych danego dnia. Wyświetlane są jedynie tytuły książek. Która z kwerend SQL posłuży do stworzenia tego raportu?

|  |
| --- |
|  |
| A. | SELECT tytul, data\_wypoz FROM ksiazki WHERE data\_wypoz = CURRDATE\_NT\_E(); |
| B. | SELECT tytul FROM ksiazki; |
| C. | SELECT \* FROM ksiazki; |
| D. | SELECT tytul FROM ksiazki WHERE data\_wypoz = CURRENT\_DATE(); |

***Wyjaśnienie poprawnej odpowiedzi:***  
Aby wygenerować raport książek wypożyczonych danego dnia, potrzebujemy wybrać tylko te rekordy z tabeli *ksiazki*, które mają datę wypożyczenia równą bieżącej dacie. Funkcja SQL CURRENT\_DATE() zwraca bieżącą datę, co pozwala na dokładne dopasowanie rekordów spełniających ten warunek. Kwerenda "SELECT tytul FROM ksiazki WHERE data\_wypoz = CURRENT\_DATE();" jest więc poprawna, ponieważ zwraca tytuły książek, które zostały wypożyczone właśnie dzisiaj. Pozostałe odpowiedzi są niepoprawne, ponieważ nie filtrują danych według daty wypożyczenia lub robią to niepoprawnie.

**PYTANIE NR 11.**

Jak zdefiniować w języku CSS takie formatowanie tabeli, żeby wiersz, na którym aktualnie znajduje się kursor myszy, zmieniał kolor tła na szary?

|  |
| --- |
|  |
| A. | tr:hover { background-color: gray; } |
| B. | tr:active { color: gray; } |
| C. | tr:active { background-color: gray; } |
| D. | tr:hover { color: gray; } |

***Wyjaśnienie poprawnej odpowiedzi:***  
Aby zdefiniować w języku CSS formatowanie tabeli, tak aby wiersz, na którym aktualnie znajduje się kursor myszy, zmieniał kolor tła na szary, należy użyć pseudoklasy :hover w połączeniu z selektorem tr, który odpowiada za wiersze tabeli. Pseudoklasa :hover jest stosowana do elementów, na które najechano kursorem myszy. Właściwość CSS background-color określa kolor tła elementu. W związku z tym, poprawna składnia CSS, która spełnia wymagania pytania to

tr:hover { background-color: gray; }

. Pozostałe odpowiedzi nie są poprawne, ponieważ dotyczą zmiany koloru tekstu (:active { color: gray; }) lub aktywacji stylu przez kliknięcie (:active { background-color: gray; }), a nie zmiany koloru tła wiersza tabeli podczas najechania kursorem myszy.

**PYTANIE NR 12.**

W języku HTML, aby zdefiniować hiperłącze otwierające się w osobnej karcie przeglądarki, należy zastosować atrybut

|  |
| --- |
|  |
| A. | rel = "external" |
| B. | target = "\_blank" |
| C. | target = "\_new" |
| D. | rel = "prev" |

***Wyjaśnienie poprawnej odpowiedzi:***  
Aby zdefiniować hiperłącze w języku HTML, które otwiera się w nowej karcie przeglądarki, należy użyć atrybutu **target** z wartością **"\_blank"**. Pozostałe odpowiedzi są niepoprawne, ponieważ atrybut **rel="external"** nie jest standardowym sposobem na otwieranie linków w nowej karcie, atrybut **target="\_new"** nie istnieje w specyfikacji HTML, a **rel="prev"** służy do określenia relacji między dokumentami, a nie do kontroli otwierania linków.

**PYTANIE NR 13.**

Tworząc tabelę w języku SQL zdefiniowano pole, którego wartości nie mogą się powtarzać. Do jego definicji należy zastosować atrybut

|  |
| --- |
|  |
| A. | IDENTITY |
| B. | DEFAULT |
| C. | UNIQUE |
| D. | NOT NULL |

***Wyjaśnienie poprawnej odpowiedzi:***  
Atrybut **UNIQUE** w języku SQL jest używany do zapewnienia, że wszystkie wartości w danej kolumnie tabeli są różne. Oznacza to, że żadne dwa wiersze nie mogą mieć tej samej wartości w tej kolumnie. Jest to zgodne z treścią pytania, która mówi o polu, którego wartości nie mogą się powtarzać. Pozostałe opcje: **DEFAULT** służy do ustawienia domyślnej wartości dla kolumny, jeśli wartość nie jest podana przy wstawianiu rekordu; **IDENTITY** jest używane do generowania unikalnych wartości, zwykle dla klucza głównego, ale nie zapewnia unikalności wartości w całej kolumnie; **NOT NULL** zapewnia, że kolumna nie może zawierać wartości NULL. Tylko **UNIQUE** bezpośrednio odnosi się do wymogu unikalności wartości w kolumnie.

**PYTANIE NR 14.**

Dana jest tabela firmy zawierająca następujące kolumny: *nazwa*, *adres*, *NIP*, *obrot* (obrót w ostatnim miesiącu), *rozliczenie*, *status*. Wykonanie kwerendy SQL SELECT sprawi, że zostaną wyświetlone

[Dana jest tabela firmy zawierająca następujące kolumny: <](https://www.testy.egzaminzawodowy.info/question.php?question_id=119144)

|  |
| --- |
|  |
| A. | jedynie nazwa oraz numer NIP firm, które w ostatnim miesiącu miały obrót mniejszy niż 4000 zł. |
| B. | wszystkie dane firm, które w ostatnim miesiącu miały obrót mniejszy niż 4000 zł. |
| C. | wszystkie dane firm, które w ostatnim miesiącu miały obrót co najmniej 4000 zł. |
| D. | jedynie nazwa oraz numer NIP firm, które w ostatnim miesiącu miały obrót co najmniej 4000 zł. |

***Wyjaśnienie poprawnej odpowiedzi:***  
Kwerenda SQL przedstawiona na zdjęciu wybiera kolumny *nazwa* i *NIP* z tabeli *firmy*, gdzie wartość w kolumnie *obrot* jest mniejsza niż 4000. Zatem wyświetlone zostaną tylko te dwie kolumny dla firm spełniających warunek dotyczący obrotu.

**PYTANIE NR 15.**

Instrukcją pętli, która jest przeznaczona do wykonania określonej liczby operacji na pewnym obiekcie lub zmiennej niebędącej tablicą, jest

|  |
| --- |
|  |
| A. | foreach |
| B. | switch |
| C. | for |
| D. | if |

***Wyjaśnienie poprawnej odpowiedzi:***  
Instrukcja pętli

for

jest przeznaczona do wykonania określonej liczby operacji na pewnym obiekcie lub zmiennej niebędącej tablicą. Pozwala ona na zdefiniowanie warunku początkowego, warunku zakończenia oraz kroku iteracji, co sprawia, że jest idealna do wykonania określonej liczby operacji. Instrukcje

switch

,

foreach

i

if

służą do innych celów:

switch

do wyboru między wieloma blokami kodu na podstawie wartości zmiennej,

foreach

do iteracji po elementach kolekcji lub tablicy, a

if

do wykonania kodu w zależności od spełnienia określonego warunku.

**PYTANIE NR 16.**

Funkcja mysqli\_num\_rows() języka PHP może być wywołana po uprzednim zastosowaniu kwerendy

|  |
| --- |
|  |
| A. | UPDATE |
| B. | INSERT |
| C. | DELETE |
| D. | SELECT |

***Wyjaśnienie poprawnej odpowiedzi:***  
Funkcja **mysqli\_num\_rows()** w języku PHP jest używana do zwracania liczby wierszy w wyniku zapytania typu SELECT. Jest to możliwe, ponieważ zapytanie SELECT służy do pobierania danych z bazy danych, co może skutkować jednym lub wieloma wierszami danych. Funkcje takie jak INSERT, DELETE, i UPDATE służą odpowiednio do dodawania, usuwania lub aktualizacji danych w bazie i nie zwracają bezpośrednio zestawu wyników, na którym można by wykonać **mysqli\_num\_rows()**. Dlatego jedynie po zapytaniu SELECT funkcja **mysqli\_num\_rows()** może być użyta do określenia liczby zwróconych wierszy.

**PYTANIE NR 17.**

Zgodnie z właściwościami ACID dotyczącymi wykonywania transakcji wymaganie izolacji (ang. *isolation*) oznacza, że

|  |
| --- |
|  |
| A. | w przypadku konfliktu z inną transakcją, obie modyfikują te same dane w tym samym czasie. |
| B. | pod pewnymi warunkami dane zmieniane przez transakcję mogą zostać wycofane. |
| C. | jeżeli dwie transakcje wykonują się współbieżnie, to zwykle nie widzą wprowadzanych przez siebie zmian. |
| D. | po wykonaniu transakcji system bazy danych będzie spójny. |

***Wyjaśnienie poprawnej odpowiedzi:***  
Właściwość izolacji w kontekście transakcji bazodanowych, będąca częścią modelu ACID (Atomicity, Consistency, Isolation, Durability), oznacza, że każda transakcja jest wykonywana niezależnie od innych transakcji i jej efekty są niewidoczne dla innych transakcji do momentu jej zakończenia. Odpowiedź "jeżeli dwie transakcje wykonują się współbieżnie, to zwykle nie widzą wprowadzanych przez siebie zmian." najlepiej oddaje tę definicję, ponieważ podkreśla izolację transakcji od siebie, co jest kluczowym aspektem zapewnienia spójności danych w systemach bazodanowych.

**PYTANIE NR 18.**

W edytorze grafiki rastrowej, aby pracować tylko na części obrazu, nie naruszając innych jego elementów, można wykorzystać

|  |
| --- |
|  |
| A. | warstwy. |
| B. | inwersję. |
| C. | skalowanie. |
| D. | kadrowanie. |

***Wyjaśnienie poprawnej odpowiedzi:***  
W edytorze grafiki rastrowej, aby pracować tylko na części obrazu, nie naruszając innych jego elementów, można wykorzystać **warstwy**. Pozwalają one na oddzielne modyfikowanie różnych segmentów obrazu bez wpływu na pozostałą część pracy. Inwersja zmienia kolory na przeciwne, kadrowanie redukuje obszar obrazu do wybranej części, a skalowanie zmienia rozmiar całego obrazu, co nie odpowiada celowi pracy tylko na części obrazu bez wpływu na inne elementy.

**PYTANIE NR 19.**

W języku PHP zastosowano funkcję

is\_float()

. Które z podanych wywołań tej funkcji zwróci wartość true?

|  |
| --- |
|  |
| A. | is\_float(NULL) |
| B. | is\_float(3.34) |
| C. | is\_float('3,34') |
| D. | is\_float(334) |

***Wyjaśnienie poprawnej odpowiedzi:***  
Funkcja

is\_float()

w języku PHP służy do sprawdzania, czy dana zmienna jest liczbą zmiennoprzecinkową. Spośród podanych opcji, jedynie

is\_float(3.34)

zawiera liczbę zmiennoprzecinkową, co oznacza, że ta funkcja zwróci wartość **true** dla tej wartości. Pozostałe opcje zawierają: wartość NULL, ciąg znaków reprezentujący liczbę (ale nie jest ona typu float) oraz liczbę całkowitą, więc dla nich funkcja

is\_float()

zwróci **false**.

**PYTANIE NR 20.**

W języku JavaScript stworzono funkcję o nazwie licz\_pitagoras, która oblicza długość przeciwprostokątnej trójkąta prostokątnego, zgodnie z twierdzeniem Pitagorasa. Funkcja pobiera dwa parametry wejściowe i zwraca wartość. Prawidłowe wywołanie takiej funkcji, wraz z pobraniem zwróconego wyniku, będzie miało postać

|  |
| --- |
|  |
| A. | c = licz\_pitagoras(a, b); |
| B. | licz\_pitagoras(a, b, c); |
| C. | licz\_pitagoras(a, b) = c; |
| D. | licz\_pitagoras(a, b); |

***Wyjaśnienie poprawnej odpowiedzi:***  
W języku JavaScript, aby przypisać wynik działania funkcji do zmiennej, używa się składni

zmienna = nazwaFunkcji(parametr1, parametr2);

. W tym przypadku funkcja *licz\_pitagoras* przyjmuje dwa argumenty (a i b), które są długościami przyprostokątnych trójkąta prostokątnego, i zwraca długość przeciwprostokątnej, zgodnie z twierdzeniem Pitagorasa. Wynik działania funkcji jest przypisywany do zmiennej *c*, co jest poprawnym sposobem pobrania i zapisania wyniku zwróconego przez funkcję w JavaScript. Pozostałe opcje są niepoprawne, ponieważ nie odpowiadają prawidłowej składni języka JavaScript dla wywołania funkcji i przypisania jej wyniku do zmiennej.

**PYTANIE NR 21.**

W przedstawionym fragmencie formularza HTML zdefiniowano pole input, o którym można powiedzieć, że

[W przedstawionym fragmencie formularza HTML zdefiniowano pole input, o któ](https://www.testy.egzaminzawodowy.info/question.php?question_id=119133)

|  |
| --- |
|  |
| A. | ma wpisany domyślny tekst "pole". |
| B. | ukrywa wprowadzone do niego znaki. |
| C. | wyświetla wprowadzone do niego znaki. |
| D. | umożliwia wpisywanie tylko wartości liczbowych. |

***Wyjaśnienie poprawnej odpowiedzi:***  
W przedstawionym fragmencie formularza HTML atrybut type="password" w elemencie input wskazuje, że pole to jest przeznaczone do wprowadzania haseł, co oznacza, że wprowadzone do niego znaki będą ukryte (zastąpione symbolami, takimi jak gwiazdki lub kropki, aby nie były widoczne na ekranie).

**PYTANIE NR 22.**

W języku CSS zapisano wspólne formatowanie dla pewnej grupy znaczników. Formatowanie takich znaczników w kodzie HTML nastąpi przez atrybut

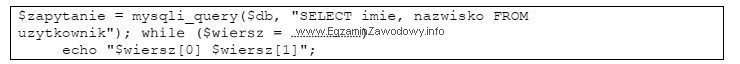
[W języku CSS zapisano wspólne formatowanie dla pewnej ](https://www.testy.egzaminzawodowy.info/question.php?question_id=119137)

|  |
| --- |
|  |
| A. | id = "format1" |
| B. | div = "format1" |
| C. | style = "format1" |
| D. | class = "format1" |

***Wyjaśnienie poprawnej odpowiedzi:***  
W języku CSS, aby zastosować wspólne formatowanie dla grupy znaczników, używa się selektora klasy, który jest oznaczony kropką (np. .format1). W kodzie HTML, aby przypisać elementowi daną klasę, używa się atrybutu class, więc poprawnym zapisem będzie class="format1". Pozostałe odpowiedzi są niepoprawne, ponieważ "div" to nazwa znacznika, a nie atrybut, "style" służy do bezpośredniego zapisu stylów dla pojedynczego elementu, a "id" służy do identyfikacji unikalnego elementu, a nie grupy elementów.

**PYTANIE NR 23.**

W prezentowanym kodzie PHP w miejscu kropek powinno znaleźć się polecenie

[](https://www.testy.egzaminzawodowy.info/question.php?question_id=119168)

|  |
| --- |
|  |
| A. | mysqli\_query($zapytanie) |
| B. | mysqli\_free\_result($zapytanie) |
| C. | mysqli\_num\_fields($zapytanie) |
| D. | mysqli\_fetch\_array($zapytanie) |

***Wyjaśnienie poprawnej odpowiedzi:***  
W prezentowanym kodzie PHP, polecenie, które powinno znaleźć się w miejscu kropek to

mysqli\_fetch\_array($zapytanie)

. Jest to funkcja służąca do pobierania wiersza z wyniku zapytania jako tablicę. Pozwala ona na iteracyjne przetwarzanie każdego wiersza zwróconego przez zapytanie SQL. Pozostałe odpowiedzi są niepoprawne, ponieważ

mysqli\_free\_result($zapytanie)

służy do zwolnienia pamięci zajmowanej przez wynik zapytania,

mysqli\_num\_fields($zapytanie)

zwraca liczbę pól w wyniku zapytania, a

mysqli\_query($zapytanie)

jest używane do wykonania samego zapytania SQL, a nie do przetwarzania wyników.

**PYTANIE NR 24.**

Które z podanych par znaczników HTML mają wizualnie to samo działanie na stronie internetowej, jeżeli żadne style CSS nie zostały zdefiniowane?

|  |
| --- |
|  |
| A. | <meta> i <title> |
| B. | <b> i <big> |
| C. | <p> i <h2> |
| D. | <b> i <strong> |

***Wyjaśnienie poprawnej odpowiedzi:***  
Znaczniki <b> i <strong> w HTML mają wizualnie to samo działanie na stronie internetowej, gdyż oba domyślnie wyświetlają tekst pogrubiony. Różnica między nimi leży głównie w semantyce, gdzie <strong> wskazuje na większe znaczenie tekstu, podczas gdy <b> służy tylko do zmiany stylu bez nadawania dodatkowego znaczenia. Pozostałe pary znaczników mają różne zastosowania i efekty wizualne: <b> i <big> (gdzie <big> jest już przestarzały i nie zalecany do użycia), <p> i <h2> (paragraf vs nagłówek drugiego poziomu, które mają różne domyślne style), oraz <meta> i <title> (służą do różnych celów w sekcji head dokumentu HTML i nie mają bezpośredniego wizualnego efektu na stronie).

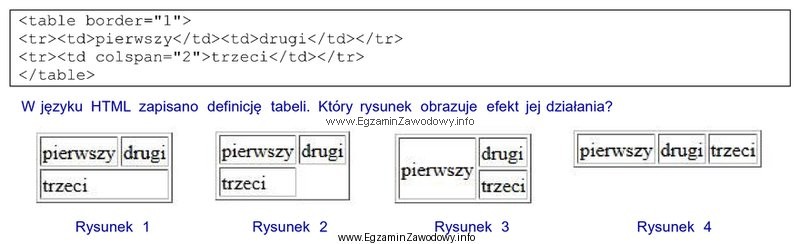
**PYTANIE NR 25.**

Język PHP posiada obsługę

|  |
| --- |
|  |
| A. | zdarzeń klawiatury. |
| B. | zdarzeń myszy. |
| C. | sesji i ciastek. |
| D. | obiektów przeglądarki. |

***Wyjaśnienie poprawnej odpowiedzi:***  
Język PHP posiada wbudowaną obsługę sesji i ciastek (cookies), co pozwala na przechowywanie danych po stronie użytkownika oraz zarządzanie sesjami użytkowników na serwerze. Sesje i ciasteczka są podstawowymi mechanizmami wykorzystywanymi do utrzymania stanu między różnymi żądaniami HTTP w aplikacjach webowych. PHP umożliwia łatwe tworzenie, odczytywanie i usuwanie ciastek, jak również zarządzanie sesjami poprzez superglobalne zmienne i funkcje dedykowane. Pozostałe odpowiedzi dotyczące zdarzeń myszy, klawiatury oraz obiektów przeglądarki nie są bezpośrednio związane z PHP, gdyż te aspekty są zazwyczaj obsługiwane przez języki skryptowe po stronie klienta, takie jak JavaScript.

**PYTANIE NR 26.**

[](https://www.testy.egzaminzawodowy.info/question.php?question_id=119132)

|  |
| --- |
|  |
| A. | Rysunek 3 |
| B. | Rysunek 2 |
| C. | Rysunek 1 |
| D. | Rysunek 4 |

***Wyjaśnienie poprawnej odpowiedzi:***  
Kod HTML definiuje tabelę z dwoma wierszami. Pierwszy wiersz zawiera dwie komórki: "pierwszy" i "drugi". Drugi wiersz zawiera jedną komórkę z atrybutem colspan="2", co oznacza, że komórka ta rozciąga się na dwie kolumny i zawiera tekst "trzeci". Rysunek 1 poprawnie przedstawia tę strukturę tabeli, gdzie pierwszy wiersz ma dwie komórki, a drugi wiersz jedną komórkę rozciągającą się na dwie kolumny.

**PYTANIE NR 27.**

W języku SQL za pomocą kwerendy ALTER można

|  |
| --- |
|  |
| A. | zmienić strukturę tabeli. |
| B. | utworzyć tabelę. |
| C. | wprowadzić dane do tabeli. |
| D. | usunąć tabelę. |

***Wyjaśnienie poprawnej odpowiedzi:***  
W języku SQL, instrukcja **ALTER** jest używana do modyfikacji struktury istniejącej tabeli w bazie danych. Może służyć do dodawania, usuwania lub modyfikowania kolumn w tabeli. Pozostałe opcje odpowiedzi: "utworzyć tabelę", "usunąć tabelę" oraz "wprowadzić dane do tabeli" są realizowane przez inne instrukcje SQL, odpowiednio: **CREATE TABLE**, **DROP TABLE** oraz **INSERT INTO**.

**PYTANIE NR 28.**

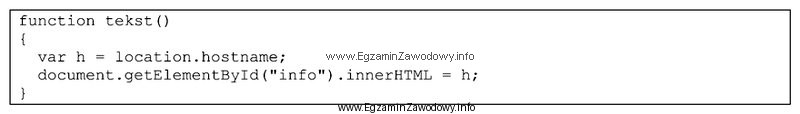
W kodzie HTML przypisano pewne znaczniki do klasy o nazwie "nomargin". Aby wykonać za pomocą języka JavaScript operacje na tych znacznikach, można posłużyć się funkcją

|  |
| --- |
|  |
| A. | getElementById("nomargin") |
| B. | getElementsByClassName("nomargin") |
| C. | getElementsByTagName("nomargin") |
| D. | getElement("nomargin") |

***Wyjaśnienie poprawnej odpowiedzi:***  
Funkcja **getElementsByClassName()** jest właściwą metodą do wybierania elementów HTML na podstawie nazwy klasy. W tym przypadku, aby wykonać operacje na znacznikach przypisanych do klasy "nomargin", należy użyć **getElementsByClassName("nomargin")**. Metoda ta zwraca kolekcję wszystkich elementów o podanej nazwie klasy. Pozostałe opcje są niepoprawne, ponieważ **getElementById()** służy do wybierania pojedynczego elementu po jego identyfikatorze, **getElementsByTagName()** wybiera elementy po nazwie tagu, a **getElement()** nie jest prawidłową metodą w JavaScript.

**PYTANIE NR 29.**

Przedstawiona funkcja, zapisana językiem JavaScript, ma za zadanie

[](https://www.testy.egzaminzawodowy.info/question.php?question_id=119165)

|  |
| --- |
|  |
| A. | wyświetlić w elemencie o id = "info" nazwę hosta, z którego pochodzi wyświetlona strona. |
| B. | wyświetlić w elemencie o id = "info" adres hosta wskazany pierwszym odnośnikiem. |
| C. | wyświetlić na przycisku lokalizację hosta, a po jego wciśnięciu umożliwić przejście do wskazanej lokalizacji. |
| D. | umożliwić przejście do wskazanej lokalizacji hosta. |

***Wyjaśnienie poprawnej odpowiedzi:***  
Analizując przedstawiony kod JavaScript, widzimy, że funkcja *tekst()* przypisuje do zmiennej *h* wartość *location.hostname*, która jest nazwą hosta bieżącej strony. Następnie wartość tej zmiennej jest wstawiana jako HTML do elementu o identyfikatorze *info*. Oznacza to, że funkcja ma za zadanie wyświetlić nazwę hosta w elemencie o id *info*.

**PYTANIE NR 30.**

Podany wpis w pliku HTML oznacza, że

[Podany wpis w pliku HTML oznacza, że](https://www.testy.egzaminzawodowy.info/question.php?question_id=119129)

|  |
| --- |
|  |
| A. | kod HTML zapisano w wersji 4 języka. |
| B. | kod HTML zapisano w wersji 5 języka. |
| C. | znaczniki zamykające są obowiązkowe w kodzie HTML, również dla znaczników samozamykających się. |
| D. | wszystkie znaczniki w kodzie HTML należy zapisywać wielkimi literami. |

***Wyjaśnienie poprawnej odpowiedzi:***  
Wpis w pliku HTML zawiera deklarację DOCTYPE dla HTML 4.01 Strict, co wskazuje na to, że dokument jest zapisany w wersji 4 języka HTML. Pozostałe odpowiedzi są niepoprawne: HTML5 ma inną deklarację DOCTYPE, wielkość liter w znacznikach HTML nie ma znaczenia (mogą być zarówno małe, jak i wielkie), a znaczniki zamykające są zazwyczaj wymagane, ale nie dla znaczników samozamykających się, które w HTML 4.01 mogą, ale nie muszą mieć dodatkowego ukośnika.

**PYTANIE NR 31.**

Dana jest tabela pracownicy, do której wpisano rekordy przedstawione obok. Po uruchomieniu podanej w ramce kwerendy SQL zostanie wyświetlona wartość

[](https://www.testy.egzaminzawodowy.info/question.php?question_id=119145)

|  |
| --- |
|  |
| A. | 2600 |
| B. | 2200 |
| C. | 5400 |
| D. | 1300 |

***Wyjaśnienie poprawnej odpowiedzi:***  
Kwerenda SQL "SELECT MAX(pensja) FROM pracownicy WHERE pensja < 3000;" zwraca największą pensję z pracowników, których pensja jest mniejsza niż 3000. W tabeli pracownicy pensje mniejsze niż 3000 to: 1300 (Monika Nowak), 2600 (Ewelina Nowakowska) oraz 2200 (Maria Kowal). Największa z tych wartości to 2600, więc jest to poprawna odpowiedź.

**PYTANIE NR 32.**

W kodzie HTML zdefiniowano formularz, który wysyła dane do pliku formularz.php. Po wciśnięciu przycisku typu submit przeglądarka przechodzi do przedstawionego adresu.  
Na podstawie podanego adresu można powiedzieć, że dane do pliku formularz.php zostały przesłane metodą

[W kodzie HTML zdefiniowano formularz, który wysyła dane ](https://www.testy.egzaminzawodowy.info/question.php?question_id=119163)

|  |
| --- |
|  |
| A. | POST |
| B. | GET |
| C. | COOKIE |
| D. | SESSION |

***Wyjaśnienie poprawnej odpowiedzi:***  
Na podstawie adresu URL widocznego na zdjęciu, który zawiera parametry (imię=Anna&nazwisko=Kowalska) przekazane po znaku zapytania (?), można stwierdzić, że dane zostały przesłane metodą GET. W metodzie GET parametry są dołączane do adresu URL i widoczne w pasku adresu przeglądarki.

**PYTANIE NR 33.**

[](https://www.testy.egzaminzawodowy.info/question.php?question_id=119135)

|  |
| --- |
|  |
| A. | czcionką pochyloną, z linią pod tekstem, wysokość wiersza na 60 px |
| B. | czcionką pochyloną, z linią nad tekstem, wysokość wiersza na 60 px |
| C. | czcionką podstawową, z linią nad tekstem, wysokość wiersza na 60 px |
| D. | czcionką podstawową, z linią pod tekstem, odległość między znakami na 60 px |

***Wyjaśnienie poprawnej odpowiedzi:***  
Na załączonym obrazie przedstawiono fragment kodu CSS, który definiuje styl dla elementów

## . Zgodnie z tym kodem, tekst będzie miał dekorację w postaci linii nad tekstem (text-decoration: overline), będzie wyświetlany kursywą (font-style: italic) oraz wysokość linii będzie wynosić 60 pikseli (line-height: 60px). Tym samym, poprawna odpowiedź to ta, która odzwierciedla wszystkie te właściwości.

**PYTANIE NR 34.**

Kolor zapisany kodem rgb(255, 10, 22), w kodzie heksadecymalnym będzie miał wartość

|  |
| --- |
|  |
| A. | #2551022 |
| B. | #FF0A16 |
| C. | #FF1016 |
| D. | #251022 |

***Wyjaśnienie poprawnej odpowiedzi:***  
Kolor w formacie RGB(255, 10, 22) należy przekonwertować na format heksadecymalny. Każda składowa RGB jest reprezentowana przez dwie cyfry heksadecymalne:  
  
Czerwony (R): 255 = FF (heks)  
Zielony (G): 10 = 0A (heks)  
Niebieski (B): 22 = 16 (heks)  
  
Łącząc te wartości, otrzymujemy kod heksadecymalny: #FF0A16. Warto zauważyć, że wartość 10 w systemie dziesiętnym jest reprezentowana jako 'A' w systemie szesnastkowym, co jest kluczowe dla poprawnego zapisu.

**PYTANIE NR 35.**

Jeśli zmienna

$x

przechowuje dowolną liczbę naturalną dodatnią, przedstawiony kod źródłowy PHP ma za zadanie wyświetlić

[](https://www.testy.egzaminzawodowy.info/question.php?question_id=119158)

|  |
| --- |
|  |
| A. | liczby wczytywane z klawiatury, tak długo aż zostanie wczytana wartość x |
| B. | kolejne liczby od x do 0 |
| C. | losowe liczby z przedziału (0, x) |
| D. | kolejne liczby od 0 do x-1 |

***Wyjaśnienie poprawnej odpowiedzi:***  
Kod PHP inicjuje zmienną $licznik wartością 0, a następnie w pętli while wyświetla wartość $licznik i inkrementuje ją o 1 w każdej iteracji, dopóki $licznik nie osiągnie wartości zmiennej $x. Oznacza to, że wyświetlane będą kolejne liczby naturalne zaczynając od 0 i kończąc na wartości o 1 mniejszej niż $x.

**PYTANIE NR 36.**

W języku JavaScript zdefiniowano obiekt. Aby dalej w kodzie modyfikować wartość właściwości x obiektu, należy zapisać

[](https://www.testy.egzaminzawodowy.info/question.php?question_id=119160)

|  |
| --- |
|  |
| A. | x = … |
| B. | wsp.x = … |
| C. | obiekt1.x = … |
| D. | obiekt1::x = … |

***Wyjaśnienie poprawnej odpowiedzi:***  
W języku JavaScript, aby zmodyfikować wartość właściwości obiektu, używa się notacji kropkowej. W podanym przykładzie obiekt nosi nazwę "obiekt1", a właściwość, którą chcemy zmodyfikować, to "x". Poprawna składnia do zmiany wartości tej właściwości to "obiekt1.x = nowa\_wartość;". Pozostałe odpowiedzi są niepoprawne: "wsp.x = …" sugeruje, że właściwość "x" należy do obiektu "wsp", który nie istnieje w podanym kodzie; "x = …" zmienia wartość zmiennej globalnej, a nie właściwości obiektu; "obiekt1::x = …" nie jest poprawną składnią w JavaScript.

**PYTANIE NR 37.**

Która z wymienionych zasad nie wpłynie korzystnie na zwiększenie czytelności kodu?

|  |
| --- |
|  |
| A. | Kod powinien być napisany bez wcięć i zbędnych enterów. |
| B. | Nazwy zmiennych powinny odzwierciedlać ich zadanie. |
| C. | Należy wprowadzać komentarze w trudniejszych częściach kodu. |
| D. | W każdej linii kodu powinna występować tylko jedna instrukcja. |

**PYTANIE NR 38.**

W języku PHP należy zapisać warunek, który będzie spełniony, gdy zmienna

$a

będzie dowolną liczbą całkowitą mniejszą od -10 lub gdy zmienna

$b

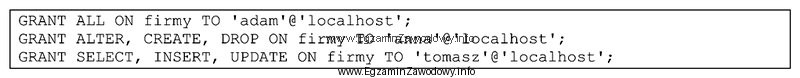
będzie dowolną liczbą z przedziału (25, 75). Wyrażenie logiczne użyte w tym warunku ma postać

|  |
| --- |
|  |
| A. | ($a < -10) and (($b > 25) or ($b < 75)) |
| B. | ($a < -10) or (($b > 25) or ($b < 75)) |
| C. | ($a < -10) or (($b > 25) and ($b < 75)) |
| D. | ($a < -10) and (($b > 25) and ($b < 75)) |

***Wyjaśnienie poprawnej odpowiedzi:***  
Warunek, który ma być spełniony, dotyczy dwóch różnych zmiennych i dwóch różnych warunków. Zmienna $a musi być mniejsza od -10, co jest wyrażone jako $a < -10. Zmienna $b musi znajdować się w przedziale otwartym (25, 75), co oznacza, że $b musi być większe niż 25 i jednocześnie mniejsze niż 75. To jest wyrażone jako ($b > 25) and ($b < 75). Ponieważ warunek może być spełniony przez spełnienie jednego z tych dwóch warunków (czyli $a jest mniejsze od -10 lub $b znajduje się w przedziale (25, 75)), używamy operatora or do połączenia tych dwóch warunków. Stąd poprawna odpowiedź to

**PYTANIE NR 39.**

W języku SQL, po wykonaniu przedstawionych poleceń GRANT, prawo do zmiany struktury tabeli oraz jej usuwania zostanie przypisane

[](https://www.testy.egzaminzawodowy.info/question.php?question_id=119151)

|  |
| --- |
|  |
| A. | Tomaszowi i Adamowi. |
| B. | tylko Annie. |
| C. | Tomaszowi i Annie. |
| D. | Adamowi i Annie. |

***Wyjaśnienie poprawnej odpowiedzi:***  
W języku SQL polecenie GRANT służy do nadawania uprawnień użytkownikom. Polecenie "GRANT ALL ON firmy TO 'adam'@'localhost';" nadaje użytkownikowi Adam wszystkie uprawnienia na tabeli firmy, co obejmuje zarówno zmianę struktury tabeli, jak i jej usuwanie. Polecenie "GRANT ALTER, CREATE, DROP ON firmy TO 'anna'@'localhost';" nadaje użytkownikowi Anna uprawnienia do zmiany struktury tabeli (ALTER), tworzenia nowych tabel (CREATE) oraz usuwania tabel (DROP). Zatem obydwoje, Adam i Anna, otrzymują prawo do zmiany struktury tabeli oraz jej usuwania.

**PYTANIE NR 40.**

Wykonanie przedstawionego polecenia PHP umożliwi

[Wykonanie przedstawionego polecenia PHP umożliwi](https://www.testy.egzaminzawodowy.info/question.php?question_id=119167)

|  |
| --- |
|  |
| A. | modyfikację danych w bazie. |
| B. | modyfikację struktury bazy. |
| C. | zapis nowych danych do bazy. |
| D. | odczyt danych z bazy. |

***Wyjaśnienie poprawnej odpowiedzi:***  
Polecenie PHP wykorzystujące funkcję mysqli\_query z instrukcją SQL "UPDATE" służy do modyfikacji istniejących danych w bazie danych. Instrukcja "UPDATE" nie jest używana do odczytu, zapisu nowych danych ani modyfikacji struktury bazy danych.