



Node MySQL/MariaDB

Ćwiczenia #1

© PIOTR SIEWNIAK 2021

Zadanie 1

Napisać program pozwalający na utworzenie na serwerze MySQL/MariaDB bazy danych o nazwie `baza_smartfony`. W bazie `baza_smartfony` należy utworzyć tabelę o nazwie `smartfony` posiadającą następujące kolumny (pola): `id`, `marka`, `model`, `rok_produkcji`, `cena`, `stan`, `nr_seryjny`.

Zrealizować obsługę ewentualnych błędów połączenia z serwerem MySQL/MariaDB oraz podczas tworzenia bazy danych i tabeli.

Zadanie 2

Napisać program pozwalający na dopisanie do tabeli `smartfony` w bazie danych `baza_smartfony` nowego rekordu zawierającego dane przykładowego smartfona. Zaimplementować kompleksową obsługę ewentualnych błędów.

Zadanie 3

Napisać program pozwalający na prezentację wybranych rekordów – danych smartfonów z tabeli `smartfony` w bazie danych `baza_smartfony` spełniających zadane kryteria, np.:

- a) `marka`;
 - b) `marka` i `model`;
 - c) `marka`, `model` i `cena`.
-

Zadanie 4

Napisać program pozwalający na modyfikację zawartości wybranych rekordów – smartfonów z tabeli `smartfony` w bazie danych `baza_smartfony`. Założyć, że modyfikowana pole to `cena`, `stan` oraz `cena` i `stan`.

Zadanie 5

Napisać program pozwalający na usunięcie wybranych rekordów – smartfonów z tabeli `smartfony` w bazie danych `baza_smartfony`. Założyć, że smartfony są usuwane na podstawie numeru seryjnego.

Zadanie 6

Założyć, że smartfony są sprzedawane w trzech komisach – filiach: w Sosnowcu, Katowicach i Gliwicach. Zaprojektować i zaimplementować kompleksową obsługę bazy danych uwzględniającą wymienione powyżej filie. Założyć, że baza danych składa się z dwóch tabel połączonych ze sobą za pomocą relacji „jeden-do-wielu”. Uwzględnić w aplikacji wszystkie operacje należące do grupy CRUD (Create, Read, Update i Delete).

UWAGI

1. Dane wejściowe:
 - a) zdefiniować w kodzie aplikacji (*wariant a*);
 - b) pobrać z pliku tekstowego (*wariant b*);
 - c) pobrać z pliku tekstowego w formacie JSON (*wariant c*).
2. Jako wyjście aplikacji wykorzystać:
 - a) konsolę (*wariant A*);
 - b) stronę WWW (*wariant B*).
3. Zaimplementować aplikacje:
 - a) bez użycia samodzielnie zdefiniowanych funkcji i klas (obiektów) (*wariant i*);
 - b) z użyciem samodzielnie zdefiniowanych funkcji i klas (obiektów) (*wariant ii*). Oprócz tego, w niezależnych – dodatkowych wariantach aplikacji wykorzystać procedury składowane (*stored procedures*).
4. W każdej aplikacji zaimplementować kompleksową obsługę błędów.
5. W kodzie źródłowym każdego z ww. programów opisać operacje realizowane przez poszczególne instrukcje za pomocą szczegółowych komentarzy.