

# Bazy danych - test 1

1. Domyślne hasło do bazy danych w XAMPP to:

- a) root
- b) db
- c) *brak hasła*
- d) mysql

2. Na jakim porcie nasłuchuje serwer MySQL?

- a) 80
- b) 8080
- c) 3306
- d) 21

3. Aby wyświetlić istniejące bazy danych, należy wydać polecenie SQL:

- a) SHOW DATABASES;
- b) GET DATABASES;
- c) PRINT DATABASES;
- d) SELECT DATABASES;

4. Aby poprawić w bazie danych imiona „Aga” na „Agnieszka”, należy wydać polecenie w SQL:

- a) CHANGE 'Aga' INTO 'Agnieszka' FROM nazwaTabeli;
- b) UPDATE nazwaTabeli SET Imie='Agnieszka' WHERE Imie='Aga';
- c) RENAME 'Aga' FROM nazwaTabeli INTO 'Agnieszka';
- d) FROM nazwaTabeli SET 'Agnieszka' WHERE Imie='Aga';

5. Aby utworzyć bazę danych za pomocą SQL, należy wydać polecenie:

- a) CREATE DB nazwaBazy;
- b) CREATE DATABASE nazwaBazy;
- c) SET DATABASE nazwaBazy;
- d) APPEND DATABASE nazwaBazy;

6. Który typ danych w SQL jest fikcyjny (nie jest prawdziwy)?

- a) varchar;
- b) char;
- c) tinychar;
- d) binary;

7. Aby usunąć tabelę, należy wydać polecenie SQL:

- a) DELETE nazwaTabeli;
- b) FROM bazaDanych DELETE nazwaTabeli;
- c) DROP TABLE nazwaTabeli;
- d) DELETE TABLE nazwaTabeli;

8. Aby wyświetlić strukturę tabeli należy wydać polecenie SQL:

- a) SHOW TABLE nazwaTabeli;
- b) SHOW nazwaTabeli;
- c) GET TABLE nazwaTabeli;
- d) DESCRIBE nazwaTabeli;

9. Aby wprowadzić dane do kolumn, należy wydać polecenie SQL:

- a) INSERT INTO nazwaTabeli [(Imie), (Nazwisko)] VALUES ('Jan', 'Kowalski');
- b) INSERT INTO nazwaKolumny (Imie, Nazwisko) VALUES ('Jan', 'Kowalski');
- c) INSERT VALUES ('Jan', 'Kowalski') INTO (Imie, Nazwisko);
- d) INTO nazwaTabeli INSERT VALUES ('Jan', 'Kowalski') IN (Imie, Nazwisko);

10. Aby wyświetlić zawartość tabeli należy wydać polecenie SQL:

- a) SHOW nazwaTabeli;
- b) FROM nazwaTabeli SHOW \*;
- c) SELECT \* FROM nazwaTabeli;
- d) SHOW \* FROM nazwaTabeli;

# Bazy danych - test 2

1. Symbol, który zastępuje dowolny ciąg znaków (tzw. *wildcard*) w SQL to:
  - a) znak \*;
  - b) znak %;
  - c) znak {\*};
  - d) znak ^;
2. Aby zmienić nazwę tabeli należy wydać polecenie w SQL:
  - a) ALTER TABLE *staraNazwaTabeli* RENAME *nowaNazwaTabeli*;
  - b) RENAME *staraNazwaTabeli* TO *nowaNazwaTabeli*;
  - c) RENAME TABLE *staraNazwaTabeli* INTO *nowaNazwaTabeli*;
  - d) CHANGE TABLE *staraNazwaTabeli* INTO *nowaNazwaTabeli*;
3. Aby usunąć wszystkie osoby o nazwisku „Nowak”, należy wydać polecenie w SQL:
  - a) DROP 'Nowak' FROM *nazwaTabeli*;
  - b) DELETE 'Nowak' FROM *nazwaTabeli*;
  - c) DELETE \* FROM *nazwaTabeli* WHERE nazwisko='Nowak';
  - d) ERASE *nazwisko* FROM *nazwaTabeli* WHERE nazwisko='Nowak';
4. Które z poniższych zdań jest prawdziwe?
  - a) tabela może zawierać maksymalnie 64 bazy danych;
  - b) każda kolumna składa się z co najmniej jednej tabeli;
  - c) w kolumnie można utworzyć tabelę za pomocą polecenia CREATE TABLE;
  - d) jedna z kolumn powinna pełnić rolę PRIMARY KEY;
5. Aby wybrać z bazy tylko mieszkańców Tychów, którzy ukończyli 18 lat, należy wydać polecenie w SQL:
  - a) SELECT \* FROM *nazwaKolumny* WHERE Miasto=*Tychy* AND Wiek >= 18;
  - b) SHOW *nazwaTabeli* WHERE Miasto='Tychy' AND Wiek >= 18;
  - c) SELECT \* FROM *nazwaTabeli* WHERE Miasto='Tychy' AND Wiek >= 18;
  - d) FROM *nazwaTabeli* SELECT \* WHERE Miasto=*Tychy* AND Wiek >= 18;
6. Aby sprawdzić ile lat liczy najstarsza osoba / osoby, należy wydać polecenie w SQL:
  - a) SELECT MAX(Wiek) AS *Najstarsza* FROM *nazwaTabeli*;
  - b) SELECT \* FROM *nazwaTabeli* WHERE Wiek=MAX();
  - c) SHOW MAX(Wiek) FROM *nazwaTabeli*;
  - d) GET MAX(Wiek) FROM *nazwaTabeli*;
7. Aby sprawdzić, ile osób mieszka w Tychach, należy wydać polecenie w SQL:
  - a) COUNT \* FROM *nazwaTabeli* WHERE Miasto='Tychy';
  - b) SELECT \* FROM *nazwaTabeli* WHERE Miasto='Tychy';
  - c) GET COUNT FROM *nazwaTabeli* WHERE Miasto='Tychy';
  - d) SELECT COUNT(%) FROM *nazwaTabeli* WHERE Miasto='Tychy';
8. Wydadź polecenie SQL, które wyświetli średnią wieku wszystkich osób:
  - a) SHOW *Wiek* FROM *nazwaTabeli* AS AVG;
  - b) SELECT AVG(Wiek) AS *Średnia* FROM *nazwaTabeli*;
  - c) COUNT *Wiek* AS AVG FROM *nazwaKolumny*;
  - d) FROM *nazwaKolumny* AVG(\*);
9. Za pomocą polecenia SQL, wybierz wszystkie osoby między 18 a 30 rokiem życia:
  - a) SELECT \* FROM *nazwaTabeli* WHERE Wiek BETWEEN 18 AND 30;
  - b) SHOW \* FROM *nazwaTabeli* WHERE Wiek=<18 AND Wiek=>30;
  - c) SELECT *Wiek* FROM *nazwaTabeli* WHERE Wiek IS (= <18 & => 30);
  - d) SELECT \* FROM *nazwaKolumny* WHERE Wiek IS NOT <18 AND >30;
10. Za pomocą polecenia SQL, dodaj kolumnę o nazwie „Plec”:
  - a) ALTER COLUMN *Plec* varchar(64) INTO *nazwaTabeli*;
  - b) UPDATE *nazwaTabeli* ADD *Plec* varchar(64);
  - c) ALTER TABLE *nazwaTabeli* ADD *Plec* char(2);
  - d) CREATE COLUMN *Plec* varchar(64) INTO *nazwaTabeli*;