

## Bazy danych NoSQL, ćwiczenie 2

temat: Baza Redis, podstawy pracy  
autor: Artur Gramacki

1. Pobierz i zainstaluj oprogramowanie Redis:
  - pobierz plik *Redis-x64-3.2.100.zip* z <https://github.com/MSOpenTech/redis/releases>,
  - wypakuj zawartość pliku ZIP do wybranego przez siebie katalogu, np *C:\redis*,
  - UWAGA 1: wersja dla Windows jest już dość stara (3.2). Jeżeli masz możliwość wykonaj ćwiczenie instalując Redis na serwerze Linux-owym (który jest najbardziej naturalnym środowiskiem pracy dla tego systemu). Aktualnie najnowsza wersja nosi numer 5.0.5 (październik 2019).
  - UWAGA 2: możesz też uruchomić Redis w środowisku Docker (obecnie też dostępny na platformie Windows, patrz [https://docs.docker.com/toolbox/toolbox\\_install\\_windows/](https://docs.docker.com/toolbox/toolbox_install_windows/) oraz <https://hub.docker.com/editions/community/docker-ce-desktop-windows>)
2. W oknie wiersza poleceń uruchom serwer Redis (*redis-server.exe*).
3. W oddzielnym oknie wiersza poleceń uruchom narzędzie klienta Redis (*redis-cli.exe*) i połącz się z uruchomionym serwerem Redis.
4. Sprawdźmy połączenie z serwerem poleceniem *ping*.
5. Wpisz jakąkolwiek pierwszą daną do bazy, odczytaj jej zawartość, zmodyfikuj ją i na końcu usuń (funkcjonalne odpowiedniki poleceń INSERT, UPDATE, DELETE, SELECT znane z klasycznych baz relacyjnych).
6. Utwórz po kilka przykładowych danych typu:
  - a. *string* (zmienna prosta),
  - b. *list* (kolekcja elementów w określonej przez wstawianie kolejności),
  - c. *set* (kolekcja unikalnych nieposortowanych elementów),
  - d. *sorted set* (kolejność elementów według ustalonej numeracji (score)),
  - e. *hash* (tablice asocjacyjne).
7. Sprawdź na zaproponowanym przykładzie działanie funkcjonalności polegającej na zmianie nazwy dowolnego polecenia oraz na możliwości zablokowania wykonywania dowolnego polecenia. Kiedy te funkcjonalności mogą być przydatne w praktyce?
8. Sprawdź na zaproponowanym przykładzie działanie funkcjonalności polegającej na możliwości ustawienia czasu życia zmiennej (polecenia TTL, EXPIRE, PTTL, oraz opcje EX, PX, NX, XX polecenia SET).
9. Sprawdź na zaproponowanym przykładzie działanie funkcjonalności obsługi transakcji (polecenia MULTI, EXEC, DISCARD, WATCH, UNWATCH).
10. Uruchom serwer (na próbę) jako usługę Windows, patrz dokumentacja: *Windows Service Documentation.docx*
11. Pobierz i zainstaluj wygodne do pracy środowisko graficzne *Redis Desktop Manager* (<https://redisdesktop.com/>)
12. Zmodyfikuj plik konfiguracyjny *redis.conf*, tak, aby serwer działał w trybie autoryzowanym. Potwierdź prawidłowe działanie tej funkcjonalności.

13. Wykonaj kopię bezpieczeństwa danych na dysku. Zatrzymaj serwer, ponownie go uruchom, a następnie sprawdź, czy wprowadzone wcześniej dane nie zostały utracone.
14. Uruchamiamy wbudowany test wydajności redis-benchmark. Posiłkując się dokumentacją (<https://redis.io/topics/benchmarks>) wykonaj kilka przykładowych testów i postaraj się dokładnie zrozumieć otrzymywane wyniki (np: liczba operacji na sekundę dla każdego z poleceń).