

## Designer to jPalio MINI-HOWTO (PL)

- ✓ W pakiecie `torn.netobjects.test` sa przyklady prostego serwera i klienta tej biblioteki, które przesyłaja między sobą różne obiekty serializowalne. Istnieja dwa typy programów klienckich: uzywajacych polaczen typu IMMEDIATE, oraz REQUEST-RESPONSE. O różnicach można przeczytać w javadoc (opis klasy `GenericConnection` i `ImmediateConnection`).
- ✓ Do jPalio można wysyłać jedynie obiekty klasy `torn.netobjects.Request`, oczekując w odpowiedzi obiektów klasy `torn.netobjects.Response`.
- ✓ Przesłanie do jPalio obiektu odpowiada wywołaniu odpowiedniej metody z któregoś z dwóch interfejsów dostępnych dla Designera: `palio.connectors.SQLConnectable`, oraz `palio.connectors.PalioConnectable`.
- ✓ Zależności pomiędzy argumentami konstruktora obiektu `torn.netobjects.Request` a wywołaniem metody w jPalio:
  - `String query` – nazwa metody do wywołania
  - `Object[] parameters` – tablica argumentów wywoływanej metody
  - `String destination` – nazwa interfejsu, z którego jest wywoływana metoda (bez poprzedzającej nazwy pakietu).
- ✓ Obiekt `torn.netobjects.Response` zawiera obiekt zwrócony przez wywoływana metodę, lub rzucony przez nią wyjątek (wtedy metoda `isSuccessful()` zwraca wartość `false`). Ponadto:
  - Metoda `getSource()` zwraca to, co było podane jako `destination` przy tworzeniu obiektu `torn.netobjects.Request`.
  - Metoda `getDestination()` zwraca to, co było podane jako `source` przy tworzeniu obiektu `torn.netobjects.Request`.
- ✓ **jPalio wymaga autorywacji polaczen.** Autoryzacja polega na porównaniu podanego loginu i hasła z tymi, przechowywanymi w tabeli `DS_USERS`. Aby połączenie zostało autoryzowane, należy, jako pierwsza, wywołać metodę `login(String login, String password)` throws `PalioException`. Nie jest ona zdefiniowana w żadnym z interfejsów jPalio.