

## Ćwiczenie - za 10 lub 10 + 6 punktów

Dla grup: wt/TP07, wt/TP09

1. Przestudiuj opis i zrealizuj w domu (**przed zajęciami wirtualnymi**) aplikację opisaną w **Ćwiczenie 5, WCF – hosting IIS, usługi typu REST, Zadanie - część I, Część A i Zadanie – część I, Część B**, plik: Cw5-WCF-3a.pdf.
2. Podczas zajęć zdalnych (labo wirtualne) zbuduj serwis WCF i klienta tego serwisu opisane w **Zadanie – Część II, punkt C**.
  - a. Wersja za 10 punktów: środowisko lokalne, klient to narzędzie **Postman**.
  - b. Wersja za 10 + 6 punktów: serwis na platformie **Azure**, klient na platformie **Azure – zobacz też Uwaga/ważne**.
3. Uwaga:
  - a. Serwis zapewnia funkcjonalności opisane w wymaganiach (**Zadanie – Część II, punkt C**) i zapewnia funkcje podręcznej biblioteki – patrz ćw. 4.
  - b. Serwis dodatkowo zapewnia operację typu **info** – zwraca datę & czas, nr albumu, imię(-na) autorów aplikacji.
    - a. Dopuszczane ograniczenie: serwis zwraca dane tylko w formacie JSON
4. Udokumentuj – w pliku *nazwisko*.docx i/lub *nazwisko*.pdf - kolejne kroki kompilacji i uruchamiania/testowania serwisu WCF i klienta tego serwisu środowisku jednomaszynowym.

**Bardzo ważne:** dokumentacja - w postaci kroków - musi zawierać krótkie komentarze/polecenia i screenshotty wprowadzonych komend i uzyskiwanych wyników. Należy również podać użytą wersję **.NET Framework** i *warunkowo* wersję narzędzia **Postman**.
5. Na zakończenie prześlij=wgraj rozwiązanie (*solution*) **Zadanie – część II, C**, na **ePortal**. Obowiązują następujące reguły:
  - wszystkie pliki mają być skompresowane do jednego pliku
  - nazwa pliku ma mieć format: **X-Nazwisko-Imie.zip** gdzie **X** - numer zadania (taki jak na witrynie),
    - bez użycia spacji - zamiast nich są myślniki (!),
    - bez polskich znaków diaktrycznych (!)np.: **3-Ziolkowski-Michal.zip**
  - wgrywany **projekt powinien być wyczyszczony** ("odchudzony")!

### **Uwaga, ważne:**

1. W wersji za 10 + 6 punktów musi być zastosowana **baza danych** dostępna jako **usługa Azure**. Baza danych musi być wstępnie wypełniona co najmniej **3 książkami żyjących autorów**. Aplikacja kliencka może być w dowolnej technologii (preferowane SPA).

2. Imię i nazwisko autora książki może zawierać wyłącznie znaki liter, pierwsza duża litera, minimum 3 litery. Gdy warunek niespełniony, to serwis zwraca błąd.
3. Pliki z kodem i załączona dokumentacja musi być tak napisana, że szybko i łatwo będę w stanie powtórzyć wszystkie kroki kompilacji, uruchamiania i testowania aplikacji.
4. Kompletne rozwiązanie (*solution*) wgrywane na ePortal musi być oczyszczone ze zbędnych plików/folderów; koniecznie usunąć podfoldery: **bin, obj, .vs**