

Zadanie – Pan Binarny

Pan Binarny założył instalację gazową w swoim samochodzie, dzięki czemu mógł wykorzystywać do jazdy 2 rodzaje paliwa. Parametry pojazdu są przedstawione w poniższej tabeli:

Paliwo	Pojemność zbiornika	Zużycie paliwa (spalanie)
PB95	45 litrów	6 litrów na 100 km
LPG	30 litrów	9 litrów na 100 km

Pan Binarny postanowił prowadzić rejestr długości pokonywanych tras w kolejnych dniach 2014 roku. Dane te zostały zebrane w pliku *lpg.txt*. Są to data wyjazdu (*data*) oraz liczba kilometrów przejechanych danego dnia (*km*). Pierwszy wiersz jest wierszem nagłówkowym, a dane w wierszach są rozdzielone znakami tabulacji.

Przykład:

```
data      km
2014-01-01 159
2014-01-02  82
2014-01-03 108
```

1 stycznia 2014 r. zbiorniki z paliwem (PB95 i LPG) były zatankowane do pełna. Codziennie rano pan Binarny decydował, w jaki sposób będzie używał paliwo:

- Jeżeli w zbiorniku LPG było więcej niż 15 litrów, to do jazdy wykorzystywał tylko paliwo LPG.
- W przeciwnym razie połowę trasy pokonywał, korzystając z paliwa PB95, a połowę – z paliwa LPG.

Liczbę zużytych litrów paliwa na trasie obliczał ze wzoru:

$$litry = \frac{spalanie * km}{100}$$

Gdzie spalanie oznacza zużycie paliwa w litrach na 100 km (zgodnie z parametrami zużycia w tabeli powyżej), a km oznacza liczbę przejechanych kilometrów, przy czym **wyniki zaokrągał do dwóch miejsc po przecinku**.

Bez względu na rodzaj paliwa tankowanie wykonywane było **po przejechaniu trasy, wieczorem, w każdy czwartek**, o ile w zbiorniku z PB95 znajdowało się poniżej 40 litrów, pan Binarny tankował do pełna PB95. Paliwo LPG było tankowane do pełna wtedy, gdy w zbiorniku LPG znajdowało się mniej niż 5 litrów tego paliwa i odbywało się **niezależnie od dnia tygodnia**.

UWAGA: Dane podane w pliku *lpg.txt* wykluczają sytuację, że panu Binarnemu kiedykolwiek zabrakło paliwa na trasie, i w związku z tym musiał on zmienić decyzję co do rodzaju używanego paliwa.

Zadania

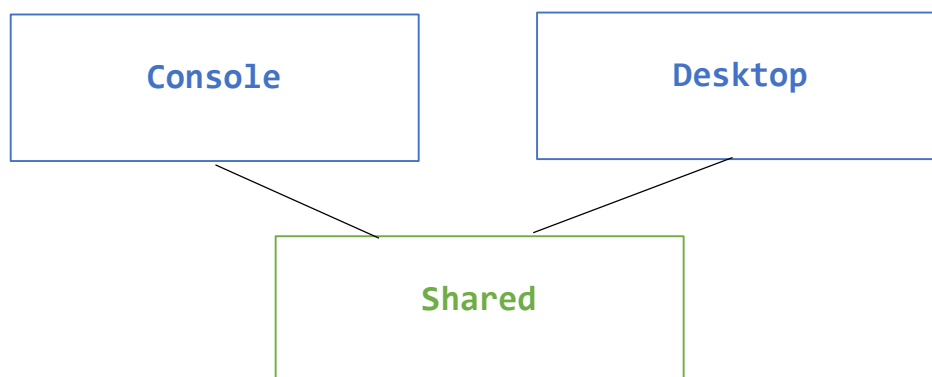
Stwórz rozwiązanie Visual Studio składające się z 3 projektów:

- Aplikacja konsoli (.NET 6.0)
- Aplikacja WPF (.NET 6.0)
- Biblioteka klas (.NET Standard 2.1)

Postaraj zachować się przy tym konwencji nazewnictwa:

- **Projekt konsoli:** `NazwaProjektu.Console`
- **Projekt WPF:** `NazwaProjektu.Desktop`
- **Projekt biblioteki klas:** `NazwaProjektu.Shared`

Dzięki temu, zarówno aplikacja konsolowa oraz mobilna będą mogły korzystać z kodu znajdującego się w utworzonej bibliotece – bez konieczności pisania go na nowo.



Biblioteka klas powinna posiadać metody które pozwolą odpowiedzieć na poniższe pytania:

1. Podaj, ile razy pan Binarny tankował paliwo PB95, a ile razy LPG.
2. Podaj liczbę dni, w których pan Binarny korzystał podczas jazdy wyłącznie z paliwa LPG.
3. Podaj dzień, w którym w zbiorniku LPG po raz pierwszy było rano mniej niż 5,25 litra paliwa.
4. Podaj stan zbiornika LPG przed podróżą oraz stan zbiornika LPG po każdej podróży.
Wygenerowane dane mają mieć możliwość zapisu do pliku w formacie JSON.
5. Przyjmijmy że w całym rozważanym okresie koszt 1 litra LPG jest równy 2,29 zł, a koszt 1 litra Pb95 wynosi 4,99 zł. Koszt instalacji gazowej to 1600 zł. **(W aplikacji WPF powinniśmy mieć możliwość samodzielnego ustalania tych wartości)**. Oblicz dla rozważanego okresu koszt eksploatacji samochodu z wykorzystaniem instalacji gazowej (uwzględnij koszt instalacji oraz zużyte LPG) oraz koszt eksploatacji samochodu wykorzystującego tylko paliwo Pb95 (w oparciu o zużytą ilość PB95). W swoich obliczeniach dotyczących eksploatacji z wykorzystaniem instalacji gazowej przyjmij wszystkie opisane założenia dotyczące tankowania i sposobu przejazdów pana Binarnego. Zauważ, że w obliczeniach związanych z wykorzystaniem wyłącznie paliwa PB95 nie ma znaczenia dzień tankowania, istotny jest jedynie koszt eksploatacji. **Podaj oba koszty eksploatacji**. Podczas obliczeń wszystkie koszty oraz wyniki końcowe zaokrąglaj do dwóch miejsc po przecinku.

Wytyczne

- Całą logikę zadania umieść w bibliotece klas – niech ona będzie odpowiedzialna za pobranie danych oraz zwrócenie odpowiedzi na poszczególne pytania. Dobrze przygotowaną bibliotekę klas będziemy mogli następnie podłączyć do aplikacji mobilnej.
- Twoje rozwiązanie najłatwiej będzie Ci testować korzystając z aplikacji konsolowej. Aplikacja konsolowa powinna składać się z prostego menu (1 – 5), po wybraniu odpowiedniej cyfry, powinna zwrócić odpowiedź na polecenie zawarte w danym zadaniu.
- Aplikacja WPF ma być jedynie graficznym interfejsem do zadania.
- Plik tekstowy umieść jako „Resource” w bibliotece klas.

Krótki komentarz do zadania

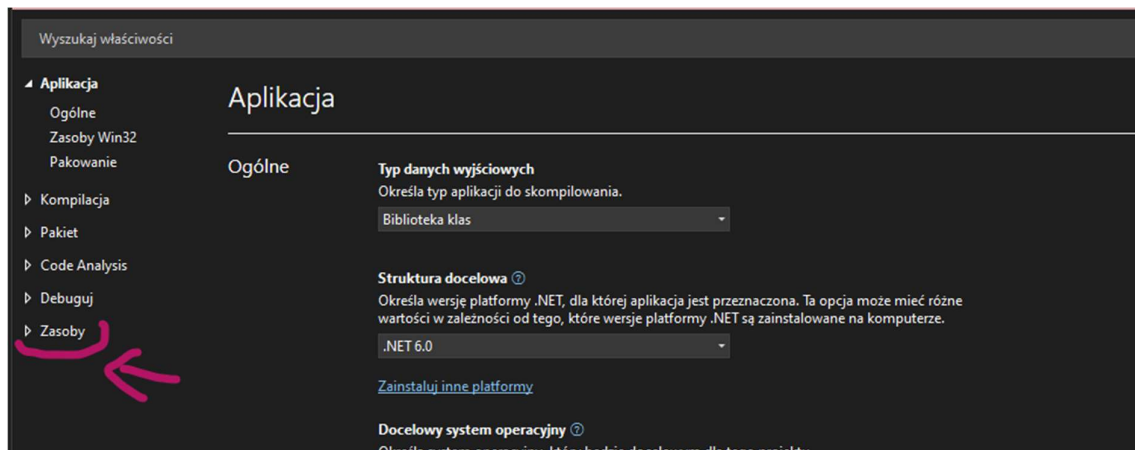
W Visual Studio należy utworzyć nowe rozwiązanie z jednym z 3 projektów. Następnie w eksploratorze rozwiązań należy kliknąć PPM na ikonkę rozwiązania i tam wybrać „Dodaj nowy projekt”. Struktura katalogów powinna być następująca:

- **NazwaRozwiązania** (katalog główny)
 - - Plik solucji (*.sln)
 - **Console**
 - - Pliki projektu konsolowego
 - Plik *.csproj – *NazwaRozwiązania.Console.csproj*
 - **Desktop**
 - - Pliki i katalogi projektu WPF
 - Plik *.csproj – *NazwaRozwiązania.Desktop.csproj*
 - **Shared**
 - - Pliki i katalogi projektu biblioteki
 - Plik *.csproj – *NazwaRozwiązania.Shared.csproj*

Następnie należy podciągnąć referencje projektu Shared do projektów Console i Desktop – poniżej znajduje się artykuł z MSDN z krótkim poradnikiem.

[Zarządzanie odwołaniami w projekcie - Visual Studio \(Windows\) | Microsoft Docs](#)

Żeby dodać plik tekstowy do projektu (dodajemy do projektu Shared) trzeba PPM kliknąć na projekt w eksploratorze rozwiązań -> *Właściwości* -> *Zasoby* (po lewej stronie)



Tam proszę utworzyć nowy plik z zasobami, i do niego dodać plik *lpg.txt*. Plik należy odczytać z zasobów:

`Properties.Resources.NazwaPliku` ← w ten sposób odwołujemy się do pliku w kodzie.

Polecam zacząć od aplikacji konsolowej jednocześnie z biblioteką – żeby na bieżąco sprawdzać czy wszystko działa. Zamysł jest taki żeby zrobić dwie aplikacje – konsolową oraz okienkową wykorzystując wielokrotnie raz napisany kod. Później podepnimy tą bibliotekę również do aplikacji mobilnej.