

## 0. Wstęp

---

Niniejsza instrukcja jest jedynie materiałem pomocniczym do warsztatu „Akademia C#” organizowanego przez kóło naukowe EKA.NET. Nie zawiera ona pełnych informacji na przedstawione tematy, jest jedynie zarysem kwestii omawianych na warsztatach i uzupełnieniem do prezentacji. Ma na celu usystematyzowanie podstawowych zagadnień, będąc wstępem do przeprowadzenia warsztatów.

Proszę o zapoznanie się z jej treścią jeszcze przed warsztatami i wykonanie kroków wypisanych poniżej, tak abyśmy mogli razem wspólnie programować podczas warsztatów, nie tracąc czasu na przygotowanie środowiska programistycznego, z którego będziemy korzystać.

- Zainstalować Visual Studio (najlepiej najnowszą wersję 2015).
- Ustawić język interfejsu VS na angielski (z takiego interfejsu będziemy korzystać podczas warsztatów, łatwiej będzie znaleźć odpowiednie funkcje, czasami tłumaczenia nazw na język polski nie są zbyt trafne. Będzie łatwiej ☺ ). Narzędzia -> Opcje -> Środowisko -> Ustawienia międzynarodowe -> Język -> English. Wymagany restart VS.
- Zainstalować dodatek do VS – ReSharper (instrukcja w punkcie 3).
- Utworzyć projekt testowy typu Hello World i skompilować (chodzi o to, aby sprawdzić czy wszystko jest dobrze zainstalowane i można spokojnie zabrać się do pracy). Instrukcja w pkt. 1.
- Zabrać ze sobą sprzęt na warsztaty (najlepiej z pełną baterią, mogą być problemy z gniazdkami na sali).
- Przyjść szczęśliwym i dobrze się bawić podczas wspólnej nauki! ☺

## 1. Tworzenie nowego projektu w Visual Studio 2015

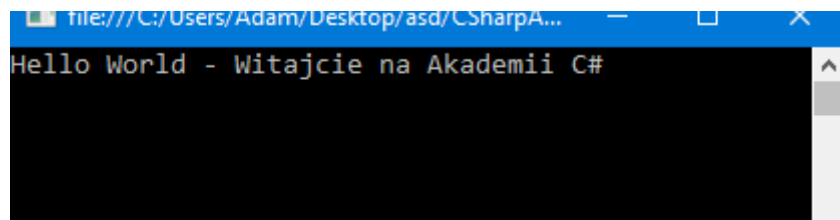
---

1. Stwórz nowy projekt w programie Visual Studio:

- a) File -> New -> Project
- b) Z lewego menu wybierz Installed -> Templates -> Visual C# -> Windows -> i następnie w środkowym oknie: Console Application
- c) Uzupełnij pola:
  - a. Name: CSharpAcademy2
  - b. Location: Dowolne
  - c. Solution name: uzupełni się automatycznie, identycznie jak „Name”
  - d. Zaznacz opcję „Create directory for solution”
- d) Nowy projekt został utworzony. W pliku Program.cs znajduje się metoda *main*. *Uzupełnij ją według poniższego wzoru:*

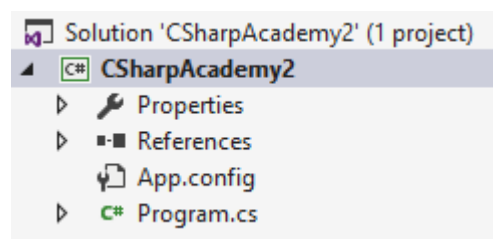
```
namespace CSharpAcademy2
{
    0 references
    class Program
    {
        0 references
        static void Main(string[] args)
        {
            Console.WriteLine("Hello World - Witajcie na Akademii C#");
            Console.ReadKey();
        }
    }
}
```

Następnie wciśnij klawisz *F5* aby debugować, zbudować i „odpalić” nasz program. Jeśli wszystko poszło dobrze, naszym oczom powinien pojawić się nasz pierwszy program.



## 2. Krótco o strukturze projektu konsolowego w Visual Studio

W panelu Solution Explorer (jeśli go nie masz, w prawym górnym rogu VS, w pole „Quick Launch”, wpisz „Solution Explorer” aby go wyświetlić). Powinniśmy widzieć następującą strukturę projektu:



### Solution vs Projekt:

- **Solution** – w jednej solucji może znajdować się wiele projektów
- **Projekt** – w naszym wypadku nazwany „CSharpAcademy2” – zawiera wszelkie potrzebne pliki do działania naszej aplikacji

W pliku *Program.cs* (.cs – rozszerzenie plików języka C#) znajduje się główna funkcja programu *main*. To od niej nasz program rozpoczyna działanie. Jest to funkcja domyślna i każdy program musi posiadać funkcję o takiej nazwie, aby został poprawnie uruchomiony.

### 3. Instalacja ReSharper'a + podstawowe skróty klawiszowe

ReSharper to jedno z najpopularniejszych i najbardziej cenionych przez programistów .NET narzędzi, które w znaczny sposób rozszerza możliwości środowiska programistycznego Microsoft Visual Studio i w ten sposób ułatwia oraz przyspiesza tworzenie bądź refaktoryzację kodu. Po instalacji ReSharper integruje się z Visual Studio i działa w tle, na bieżąco oferując podpowiedzi i zalecenia daleko wykraczające poza to, co Visual Studio i jego IntelliSense posiada w standardzie. Program pozwala także między innymi jednym kliknięciem refaktoryzować i przenosić duże fragmenty kodu, a także kompiluje pisany kod "na żywo" od razu wyłapując ewentualne błędy.

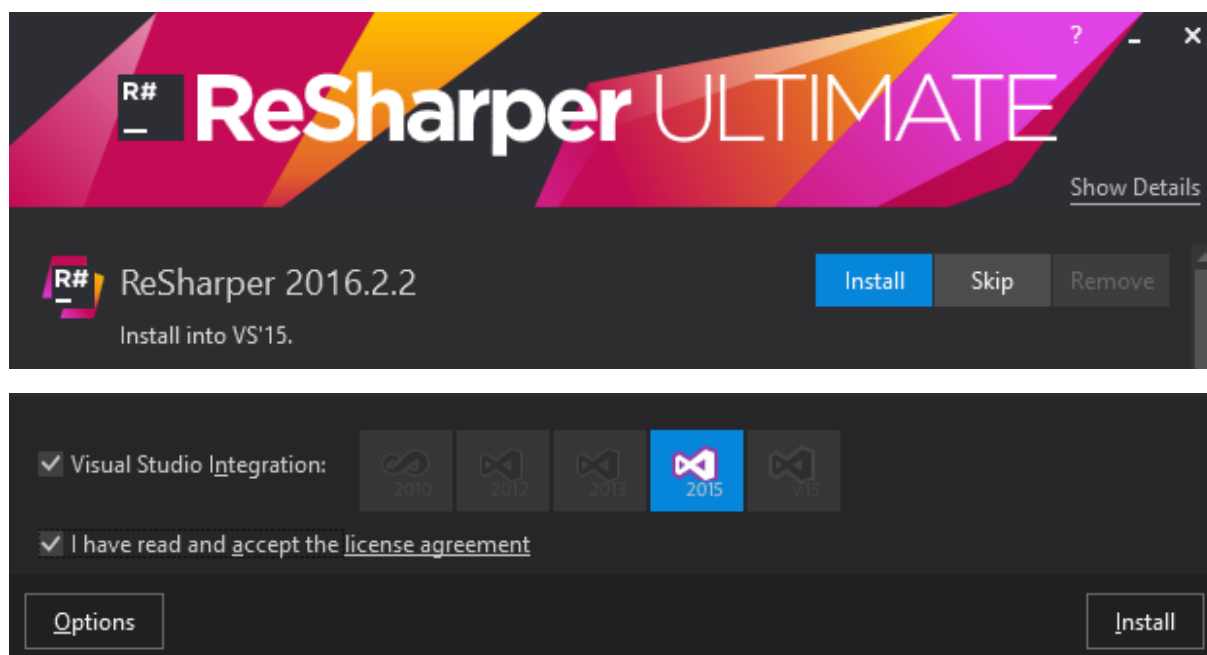
Generalnie ReSharper jest narzędziem płatnym, jednak dla studentów Politechniki Wrocławskiej jest on całkowicie darmowy!

1. Wchodzimy na stronę:

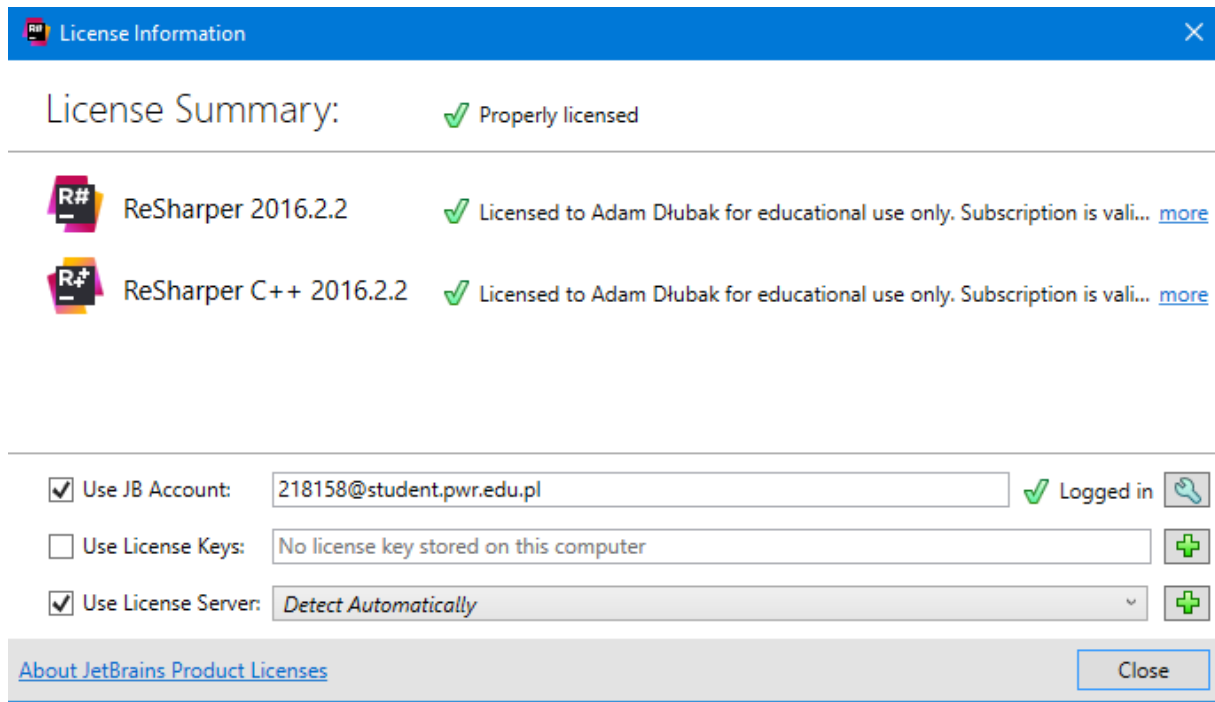
<https://www.jetbrains.com/resharper/download/download-thanks.html?code=RSU&platform=windows>

I pobieramy ReSharper'a – reprezentowany jest on tam jako 30-dniowy trial jednak podczas instalacji wystarczy przedstawić się jako student PWr aby otrzymać pełną licencję.

2. Uruchom pobrany plik instalacyjny i wybierz opcję:

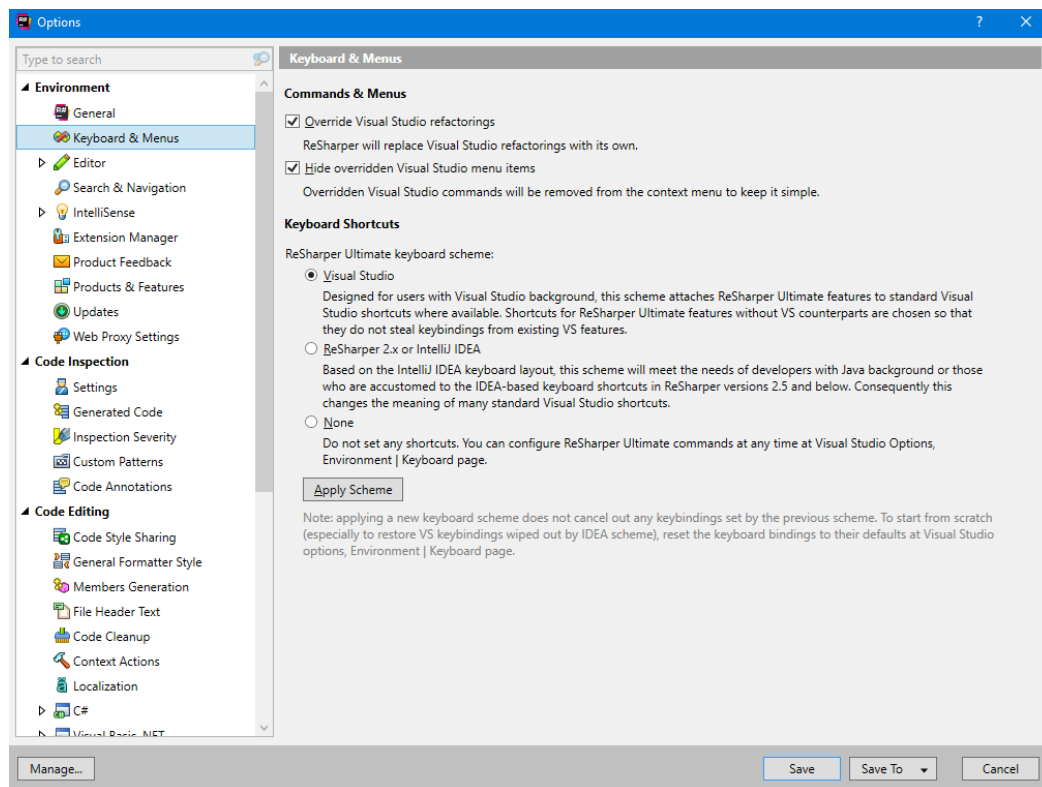


A następnie zainstaluj. Po wszystkim uruchom VS i Twoim oczom powinien pokazać się mniej więcej taki widok (jeśli nie, wybierz -> ReSharper (z górnego menu) -> Help -> License Information)



Musicie użyć swojego nowego Use JB Account aby otrzymać licencję (login to nasz adres mailowy z Politechniki).

Następnie wchodzimy w ReSharper -> Options i w pierwszym oknie sprawdzamy czy mamy ustawione wszystko jak na poniższym screenie (konfiguracja ta umożliwia korzystanie ze skrótów klawiszowych ReSharpera).



Wspomnianą wcześniej rozpiskę skrótów klawiszowych znajdziecie w poniższym linku. Polecam wydrukować, powiesić nad biurkiem i korzystać kiedy tylko się da!

[https://www.jetbrains.com/resharper/docs/ReSharper\\_DefaultKeymap\\_VSscheme.pdf](https://www.jetbrains.com/resharper/docs/ReSharper_DefaultKeymap_VSscheme.pdf)

Ciekawsze i przydatniejsze skróty na najbliższe warsztaty:

- |                      |  |
|----------------------|--|
| • Alt + Enter        | Show available quick-fixes and context actions |
| • Ctrl + Alt + Enter | Reformat code                                  |
| • Alt+Ins            | Generate code (constructors, properties, etc)  |
| • Ctrl+D             | Duplicate a line or selection                  |
| • Ctrl + /           | Comment/uncomment line                         |
| • Ctrl+R, R          | Rename   |
| • Ctrl+T             | Go to everything                               |

## 4. Zarys aplikacji do przygotowania podczas warsztatów

---

Aby powtórzyć wszystkie zagadnienia omówione podczas warsztatów, jako ich podsumowanie przygotowujemy prostą aplikację mierzącą czas.

Należy przygotować aplikację konsolową, która:

- Będzie zawierała menu w następującej postaci:
  - Pokaż aktualną datę i godzinę
  - Stoper
  - Minutnik

O. Wyjdź z aplikacji

- Menu główne ma być wyświetlane w nieskończonej pętli while
- Wybór opcji ma dokonywać się w instrukcji wyboru switch
- Aby pobrać znak od użytkownika należy użyć sekwencji:  
`Console.ReadKey().KeyChar;`
- Aby wyczyścić okno konsoli należy użyć:  
`Console.Clear();`
- Implementacja każdej opcji menu ma znajdować się w osobnej metodzie.
- Minutnik ma odmierzać czas w sekundach, których ilość podaje użytkownik w konsoli i jest ona przekazywana jako parametr do funkcji. (należy użyć `Int32.Parse` aby przekonwertować wartość podaną przez użytkownika na typ int). Minutnik

ma odliczać czas w pętli for, aby zatrzymać działanie programu na czas jednej sekundy należy użyć polecenia `System.Threading.Thread.Sleep(1000);`

- Stoper na nasze warunki będzie pracował w następujący sposób:
  - Użytkownik podobnie jak w przypadku minutnika podaje czas do odliczania, a on przekazywany jest jako parametr do funkcji.
  - Odliczanie odbywa się w pętli do while, co oznacza, że stoper odliczy co najmniej jedną sekundę (wybieramy ten konkretny rodzaj pętli, aby wykorzystać wszystkie poznane przez nas metody (w normalnych warunkach można wykonać to w inny sposób)).
- Aby wyświetlić aktualną datę i godzinę należy użyć:

`DateTime.Now`