LEKCJA 2 – Zmienne i stałe

Zaczynamy tworzenie naszej aplikacji konsolowej.

# Utworzenie nowego projektu

Tworzymy nowy projekt (*Create a new project*) aplikacji konsolowej .NET Core (*Console App (.NET Core)*). Aby zawęzić opcje wyszukiwania odpowiedniego typu aplikacji możemy ustawić język wyszukiwania (rozwijane menu z napisem *All languages*) wybierając opcję C#. Następnie możemy ustawić platformę, na której ma działać nasz program (rozwijane menu z napisem *All platforms*) jako Windows, oraz wybrać typ projektu (rozwijane menu z napisem *All project types*) – aplikacja konsolowa (*Console*). Przy wyborze typu projektu należy zwrócić uwagę, aby wybrać opcję z językiem C#, a nie np. Visual Basic (VB). Następnie nadajemy nazwę naszemu projektowi. Przypominamy, że nazwa powinna być w formacie PascalCase (wyrazy pisane łącznie, pierwsza litera każdego wyrazu wielka, nazwa najlepiej w języku angielskim). Jeżeli chcemy możemy zmienić lokalizację projektu (*Location*), aby później łatwo było nam go znaleźć. Tworzymy nasz projekt. Możemy go od razu umieścić na naszym koncie na githubie, tworząc nowe repozytorium.

# Umieszczenie projektu na naszym koncie na githubie

## Za pośrednictwem Visual Studio

Można to zrobić np. poprzez kliknięcie zakładki *Git* z menu głównego i wybranie opcji *Create Git Repository*. Następnie uzupełniamy dane w oknie *Create a Git Repository*, które nam się pojawi. Zmieniamy w nim typ repozytorium z prywatnego (odznaczamy opcję *Private repository*, jeżeli jest zaznaczona) na publiczny (dostępny dla wszystkich).

## Za pośrednictwem strony internetowej

Alternatywnie możemy utworzyć nowe repozytorium na swoim koncie przez [stronę internetową](https://github.com/login), a następnie umieścić w nim nasz projekt.

1. Zaloguj się na swoim koncie github.
2. Kliknij przycisk *New* przy *Repositories*.
3. Wpisz nazwę repozytorium (*Repository name*), najlepiej taką samą jak nazwa naszego projektu. Jeśli chcemy możemy dodać opis projektu (*Description*). Pozostawiamy wybraną opcję *Public* (tworzymy repozytorium publiczne – dostępne dla wszystkich).
4. Można również dodać plik *README*, zaznaczając opcję *Add a README file*, w którym możemy opisać nasz projekt.
5. Od razu możemy dodać też plik .*gitignore* zawierający pliki będące częścią naszego projektu, które nie chcemy jednak, aby zostały umieszczone w naszym repozytorium. Możemy wybrać gotowy szablon pliku (*.gitignore template*). Nie ma na razie szablonu dla platformy .NET Core. Możemy więc wybrać szablon *Visual Studio* lub utworzyć pusty plik (typ szablonu *None*). Utworzony plik można później zmodyfikować do naszych potrzeb.

Można również wykonać te czynności za pośrednictwem konsoli Windows, poprzez użycie odpowiednich poleceń (pogrubiony tekst należy zastąpić właściwymi danymi):

cd **ścieżka\do\folderu\z\projektem**

git init

dotnet new gitignore

git add -A

git commit -m "Initialize project"

Jeżeli nasza konsola nie obsługuje poleceń *git*, to być może nie mamy go jeszcze zainstalowanego w naszym systemie. Należy go wówczas pobrać i zainstalować zgodnie ze wskazówkami ze [strony](https://github.com/git-guides/install-git). Dotneta zainstalowaliśmy razem z programem Visual Studio, ale jeżeli nasza konsola nie obsługuje polecenia *dotnet*, być może musimy dodać jego lokalizację do *Path* w zmiennych środowiskowych (Klikamy ikonkę Windows i wyszukujemy *Edytuj zmienne środowiskowe systemu*. -> Po kliknięciu na wynik wyszukiwania pojawi nam się okno *Właściwości* systemu, zakładka *Zaawansowane*. -> Klikamy *Zmienne środowiskowe...* -> W *Zmienne systemowe* wybieramy zmienną *Path*, *path* lub *PATH* i klikamy *Edytuj...* -> Klikamy *Nowy*. -> Wpisujemy ścieżkę dostępu do folderu zawierającego plik *dotnet.exe*, np. *C:\Program Files\dotnet*. - > Na zakończanie klikamy *OK*.). Jeżeli nasza konsola nie obsługuje poleceń *git*, pomimo, że jest on zainstalowany w systemie to być może również trzeba dodać go do *Path* w zmiennych środowiskowych, analogicznie jak dotnet. Kiedy już wszystko zainstalujemy i wykonamy powyższe polecenia, to znaczy, że utworzyliśmy już lokalne repozytorium gitowe i umieściliśmy w nim nasz projekt. Teraz musimy wejść na nasze konto github na [stronie internetowej](https://github.com/login) i utworzyć nowe puste repozytorium, najlepiej o tej samej nazwie co nasz projekt. Następnie możemy wrócić do naszej konsoli i wykonać kolejne polecenia:

git remote add origin **url/utworzonego/przed/chwilą/ repozytorium/na/naszym/koncie/github**

git push --set-upstream origin master

Po ich wykonaniu aktualna wersja naszego projektu została umieszczona w repozytorium na naszym koncie github. Być może będziemy musieli ustawić opcje w globalnej konfiguracji git, takie jak nazwa użytkownika (autora), czy e-mail:

git config --global user.name **NazwaUżytkownika(naszPodpis)**

git config --global user.email **NaszAdresEmail**

Powyższe polecenia ustawią nazwę użytkownika i email, które będą stosowane dla wszystkich repozytoriów utworzonych przez aktualnie zalogowanego użytkownika systemu. Można również ustawić te właściwości tylko dla danego repozytorium, wówczas zamiast *--global*, piszemy *--local*.

Alternatywnie możemy utworzyć nowe repozytorium i umieścić w nim nasz projekt przy pomocy przeznaczonej do tego aplikacji, takiej jak np. *GitHub Desktop* czy *Git GUI*.