

Zajęcia 3: Dodanie do projektu klasy **Pracownik**

1. Opracowanie klasy **Pracownik**

Tworzenie szkieletu klasy

Utwórz nowy projekt wybierając opcję **New|Project** z menu **File** (tak jak na zajęciach 2). Do katalogu nowo utworzonego projektu skopiuj pliki z rozszerzeniem `cpp` i `h` z poprzednich zajęć laboratoryjnych. A następnie z menu **Project|Add Existing Item...** (lub skrótem Lewy Shift+Alt+A) dołącz pliki `Data.cpp`, `Data.h`, `Napis.cpp`, `Napis.h`. Teraz przełącz panel z zawartością projektu w tryb widoku klas (**ClassView**) i dodaj do projektu kolejną klasę o nazwie `Pracownik` (postępuj analogicznie jak w przypadku dodawania klasy `Data` i `Napis` (patrz instrukcja z zajęć 2)). W widoku klas będą już widoczne trzy dodane do projektu klasy. Po powrocie do widoku projektu (**Solution Explorer**) zauważysz, że utworzone zostały dwa pliki **Pracownik.cpp** i **Pracownik.h**.

Dodanie danych składowych do klasy **Pracownik**

W definicji klasy (plik **Pracownik.h**) dodaj z użyciem kreatora następujące prywatne pola składowe:

```
Napis m_Imie;
Napis m_Nazwisko;
Data m_DataUrodzenia;
```

Następnie w części publicznej klasy zdefiniuj:

- metody informacyjne:


```
const char* Imie() const;
const char* Nazwisko() const;
```

W polu **Return type** nie ma zwracanego typu `const char*` należy, zatem wpisać go ręcznie;
- metody składowe ustawiające poszczególne składniki:


```
void Imie(const char* nowe_imie);
void Nazwisko(const char* nowe_nazwisko);
void DataUrodzenia(int nowy_dzien, int nowy_miesiac, int nowy_rok);
```
- metodę wyprowadzającą dane składowe do standardowego strumienia wyjściowego (`cout`):


```
void Wypisz() const;
```

Dane mają być wyprowadzane w jednej linii w postaci:

```
imie      nazwisko      dzien-miesiac-rok
```
- metodę wprowadzającą dane składowe ze standardowego strumienia wejściowego (`cin`):


```
void Wpisz();
```
- metody porównujące wartość pola z przekazywanym wzorcem:


```
int SprawdzImie(const char* por_imie) const;
int SprawdzNazwisko(const char* por_nazwisko) const;
```

metody mają zwracać wartość 0, gdy składnik klasy (`m_Imie/m_Nazwisko`) jest identyczny z podanym wzorcem (`por_imie/por_nazwisko`), wartość większą od zera, gdy pole (`m_Imie/m_Nazwisko`), jest alfabetycznie dalej niż przekazanego argumentu (`por_imie/por_nazwisko`) i mniejszą od 0 w przeciwnej sytuacji.

2. Opracowanie funkcji głównej (**main**)

W funkcji głównej należy zademonstrować działanie poszczególnych metod opracowanych klas z instrukcji drugiej i trzeciej najlepiej do tego użyć menu z wyborem poszczególnych opcji.

Uwaga:

Jeśli przy tworzeniu projektu wybrałeś **Empty project** to do danego projektu konieczne jest dołączenie jeszcze jednego pliku o rozszerzeniu **.cpp**. W tym celu (będąc w widoku klas) z menu **Project** należy wybrać opcję **Add New Item...**, w okienku rozwinąć katalog **Visual C++** (domyślnie), następnie zaznaczyć opcję **Code**, i kliknąć **C++ File (.cpp)**. Na koniec podać nazwę dołączanego pliku w polu **Name:** (np. **program**) i zatwierdzić wciskając przycisk **Add**.

Funkcję główną: `int main()` należy umieścić właśnie w tym pliku. Będzie ona widoczna zarówno w folderze **Global Functions and Variables** w widoku klas jak i folderze **Source Files** w widoku projektu. Nie zapomnij o dołączeniu pliku nagłówkowego **Data.h**, **Napis.h** oraz **Pracownik.h**.