Języki i Paradygmaty Programowania sem.4

# Laboratorium 2.

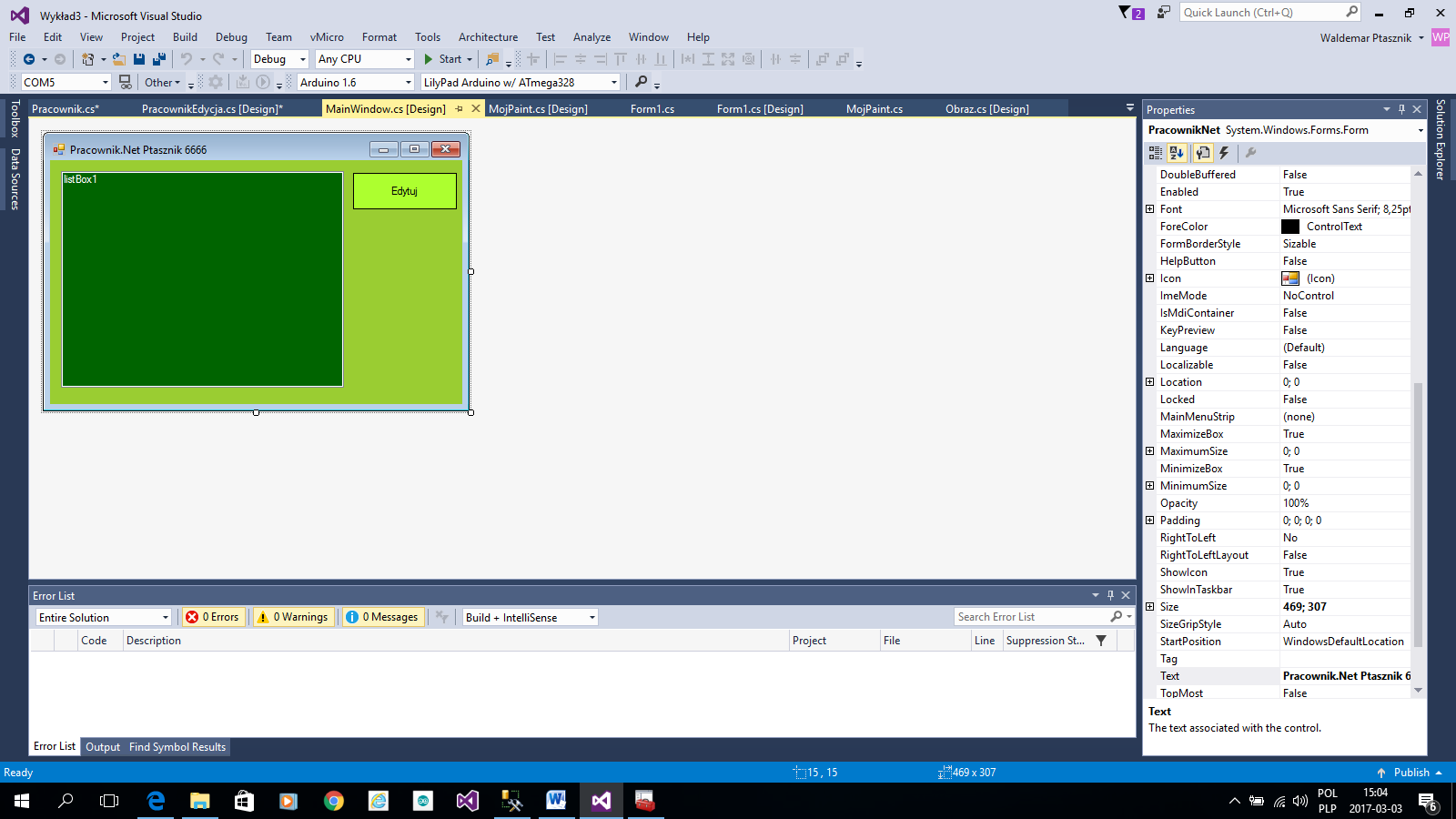
## Budowanie graficznego interfejsu użytkownika, obsługa interakcji z użytkownikiem

## Zadanie

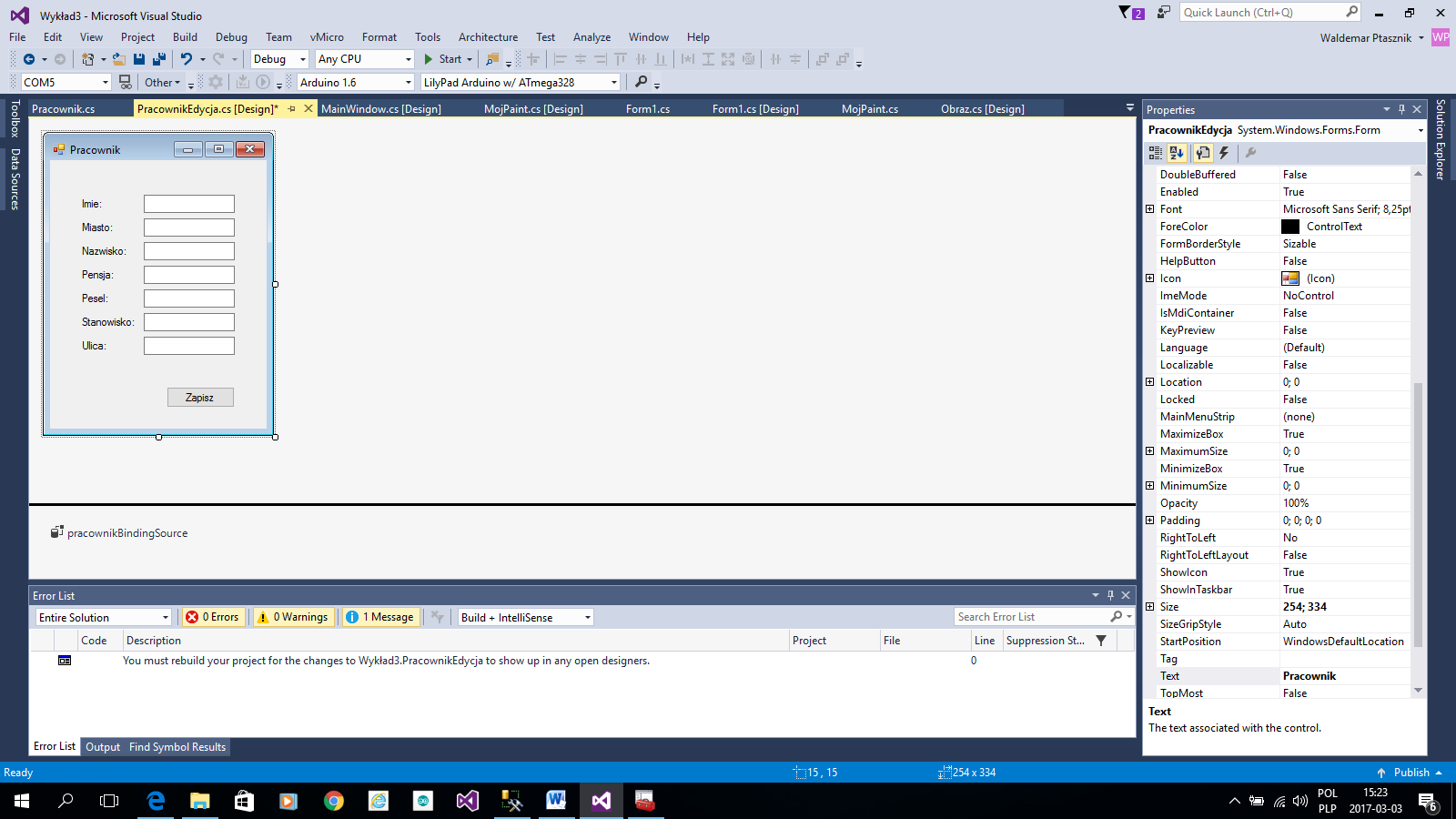
Utwórz aplikację do zarządzania danymi pracowników w przedsiębiorstwie.

Aplikacja ma składać się z okna głównego w którym wyświetlana będzie lista pracowników i okna do edycji danych pracownika, które uruchamia się po wyborze pracownika z listy i wejściu w tryb edycji za pomocą przycisku.

1. Utwórz aplikację typu WindowsForms
2. Dodaj nową klasę **Pracownik** o następujących
   1. właściwościach
      1. Imię
      2. Nazwisko
      3. Pesel
         1. Zabezpiecz właściwość aby przyjmowała wartości 11 znakowe i tylko cyfry
      4. Ulica
      5. Miasto
      6. Stanowisko
      7. Pensja
   2. Metodach
      1. Publiczna Zapisz – metoda ma zapisywać wszystkie dane osoby do pliku o nazwie [Pesel].txt
      2. Publiczna statyczna Otworz(string pesel) – metoda tworzy obiekt Pracownik z danych pobranych z pliku i zwraca jego instancję.
         1. otwiera plik o nazwie zgodnej z podanym peselem [pesel].txt
         2. tworzy obiekt Pracownik
         3. uzupełnia dane
         4. zwraca utworzony obiekt
      3. **publiczna nadpisana metoda wirtualna ToString() zwracająca imię i nazwisko pracownika**
3. Zmodyfikuj formularz główny
   1. Zatytułuj główny formularz „Pracownik.NET [Twoje Nazwisko i numer indeksu] ” (Właściwość Text) – przykład: „Pracownik.Net Ptasznik 6666”
   2. Zmień nazwę klasy formularza i nazwę pliku klasy na MainWindow(Warto zacząć od pliku to nazwa klasy sama się zmieni)
   3. Zmień tło formularza na wybrany przez siebie kolor



1. Znajdź w oknie Solution Explorer plik **AssemblyInfo.cs** I zmodyfikuj atrybut **AssemblyCopyright** podając nazwę taką samą jak w pkt. 3.a – Ta nazwa będzie widoczna w skompilowanym pliku .exe w jego właściwościach
2. Dodaj pole listy [ListBox] w którym będzie przechowywana i wyświetlana lista pracowników
3. W zdarzeniu **Load** formularza głównego zaimplementuj dodanie trzech pracowników do listy „ListBox”
   1. Utwórz obiekt Pracownika
   2. Uzupełnij przykładowymi danymi
   3. Dodaj do listy
      1. Dodawanie do listBox’a : listbox1.Items.Add(nowyPracownik);
   4. Powtórz czynność 3 razy dla każdego pracownika
4. Dodaj nowy formularz zatytułowany Edycja Pracownika



* 1. Nazwa klasy formularza: EditEmployeeWindow
  2. Dodaj pola edycyjne [TextBox] i etykiety [Label] opisujące dane klasy Pracownik
  3. Zaproponuj własny układ pól i kolorystykę
  4. Utwórz publiczną metodę **Edytuj(Pracownik pracownik)**
     1. Zaimplementuj uzupełnienie pól formularza danymi z przekazanego obiektu **pracownik**
  5. Utwórz przycisk Zapisz
     1. W zdarzeniu Click przycisku zaimplementuj przepisanie danych z pól formularza do obiektu **pracownik** i zamknięcie bieżącego formularza. **(this.Close())**

1. Dodaj przycisk **Edytuj** po prawej stronie listy pracowników w oknie głównym
2. W obsłudze kliknięcia przycisku **Edytuj**
   * 1. Zaimplementuj utworzenie nowego formularza typu EditEmployeeWindow

EditEmloyeeWindow edycja=new EditEmployeeWindow();

* + 1. Pobierz zaznaczony element listy pracowników i przekaż jako parametr metody **Edytuj.** Wywołaj metodę.

edycja.Edytuj(listBox1.SelectedItem as Pracownik);

(**SelectedItem** zwraca zawsze **Object** natomiast my mamy pewność, że dodawaliśmy do listy obiekty **Pracownik**, więc możemy śmiało wykonać rzutowanie na typ **Pracownik**)

* + 1. Sprawdź przed wywołaniem czy na pewno zaznaczono jakiś element listy
    2. Pokaż utworzone okno **EditEmployeeWindow**

edycja.ShowDialog();

1. Sprawdź działanie programu
2. Spakuj program i prześlij do zadania Laboratorium 2 na platformie Moodle