

# Warsztaty MVC 1, Dzień

5

v3.0

# Plan

## 1. Warsztaty

# Warsztaty

# Cel Warsztatów

Cel warsztatów to stworzenie systemu będącego internetową książką telefoniczną.

Aplikacja musi być oparta o framework ASP.NET MVC Core.

Aplikacja powinna przechować dane w bazie danych.

Aplikacja musi posiadać następujące funkcje:

- Dodawanie, edycja i usuwanie osoby z systemu.
- Wyświetlanie listy osób.
- Wyszukiwanie osób w systemie po nazwisku.

Opcjonalnie lista osób oraz wyniki wyszukiwania powinny obsługiwać stronicowanie.

# Repozytorium

Utwórz repozytorium w serwisie Github.

Podczas tworzenie nowego repozytorium wybierz plik `gitignore` dla programu Visual Studio.

Pobierz repozytorium na dysk i utwórz w nim projekt ASP.NET Core Web Application.

# Model Danych

Utwórz model danych o nazwie `PersonModel`.

Model musi przechowywać następujące informacje.

- Identyfikator - ID.
- Imię - `FirstName`.
- Nazwisko - `LastName`.
- Telefon - `Phone`.
- Email - `Email`.
- Data Utworzenia - `Created`.
- Data Modyfikacji - `Updated`.

Dobierz odpowiednie typy danych.

Każda właściwość musi być opatrzona odpowiednimi atrybutami, aby spełniać odpowiednie warunki walidacji oraz posiadać odpowiednie opisy.

Od Ciebie zależy jakie warunki walidacji dobierzesz dla poszczególnych właściwości.

Model utwórz w katalogu `Models`.

# Baza Danych

Ponieważ projekt będzie oparty o bazę danych, właśnie od struktury bazy zaczniemy projektowanie aplikacji.

Otwórz program SQL Management Studio i utwórz nową bazę danych.

W bazie utwórz tabelę o nazwie People.

W tabeli utwórz kolumny odpowiadające modelowi PersonModel.

# MVC

Włącz w projekcie obsługę wzorca MVC poprzez odpowiednią konfigurację w pliku `Startup`.

Utwórz odpowiednią strukturę katalogów: `Controllers`, `Views`, `Shared`.

Utwórz kontroler o nazwie `PersonController`.

Utwórz plik layoutu, pliki `ViewStart` i `ViewImports`.

Ustaw layout dla wszystkich widoków.

Włącz obsługę TagHelperów. W pliku layoutu utwórz podstawową strukturę html.



# Dostęp do Danych

Ponieważ dane w programie przechowywane są w postaci obiektów, a w bazie w formie wierszy tabeli, musimy stworzyć klasę, która pozwoli mapować jedną postać na drugą.

Klasa będzie pozwalać na wygodny dostęp do danych i pozwoli nie implementować mapowania w poszczególnych akcjach.

Utwórz klasę o nazwie `SourceManager`.

Klasa `SourceManager` będziemy używać do zapisywania, aktualizowania i pobierania danych z bazy.

# Metody SourceManager

Klasa SourceManager powinna zawierać następujące metody:

- `List<PersonModel> Get(int start, int take)` - metoda powinna zwracać listę obiektów z bazy zaczynającą się od podanego numeru wiersza. Długość listy określa drugi parametr.
- `PersonModel GetByID(int id)` - metoda powinna zwracać osobę o podanym ID.

- `int Add(PersonModel personModel)` - metoda powinna zapisać dane obiektu do bazy danych, a zwrócić automatycznie nadany identyfikator osoby.
- `void Update(PersonModel personModel)` - metoda powinna aktualizować podaną osobę w bazie danych.
- `void Remove(int id)` - metoda powinna usuwać podaną osobę z bazy danych.

Możesz dodać również swoje metody, jeśli znajdzie taka potrzeba.

# Modyfikacja Routingu

Domyślny routing w razie niepodania jawnie nazwy kontrolera będzie próbował wywołać kontroler o nazwie Home.

W naszym przypadku głównym kontrolerem jest `PersonController`.

Zmień routing tak, aby domyślnym kontrolerem był kontroler `Person`.

Dodatkowo utwórz drugą regułę routingu, która zamiast parametru, `id`, będzie przyjmowała parametr o nazwie `page`.

Drugi routing musi wyglądać następująco:  
`{controller=Person}/{action=Index}/{page?}`

# Lista Osób

W kontrolerze utwórz akcję Index, która zwraca widok.

Akcja powinna zwracać widok z listą osób pobranych z bazy danych.

# Lista Osób - Stronicowanie

Wyświetlanie wpisów powinno obsługiwać stronicowanie.

Każda strona powinna wyświetlać określoną ilość wpisów i zawierać linki do do innych stron.

W akcji Index utwórz parametr o nazwie page typu `int`.

Parametr powinien mieć domyślną wartość 1 i będzie zawierał numer strony, która ma zostać wyświetlona.

Do Ciebie należy decyzja ile elementów wyświetlisz na stronie. Jeśli stronicowanie wydaje się dla Ciebie zbyt skomplikowane, wróć do niego na końcu.

# Dodawanie Osób

Utwórz akcję Add typu get, która zwraca widok.

Utwórz widok dla akcji Add. W widoku utwórz formularz na podstawie modelu `PersonModel`.

Formularz, który znajduje się w widoku akcji Add, musi zostać wysłany na akcję Add typu post.

Utwórz drugą akcję o nazwie Add typu post, która w parametrze przyjmuje model `PersonModel`.

W akcji typu post sprawdź validację i zapisz dane z formularza do bazy danych.

W widoku Index, utwórz link kierujący na akcję Add.

# Edytowanie Osób

Utwórz akcję `Edit` typu `get`.

W akcji utwórz parametr o nazwie `id` typu `int`, na podstawie, którego pobierz osobę z bazy i przekaż do widoku.

Utwórz widok dla akcji `Edit`. W widoku utwórz formularz na podstawie modelu `PersonModel`.

Formularz, który znajduje się w widoku akcji `Edit`, musi zostać wysłany na akcję `Edit` typu `post`.

Utwórz drugą akcję o nazwie `Edit` typu `post`, która w parametrze przyjmuje model `PersonModel`.

W akcji typu `post` sprawdź walidację. Akcji musi uaktualniać dane osoby na podstawie jego identyfikatora.

W widoku `Index`, przy każdym wpisie utwórz link kierujący na akcję `Edit`.



# Usuwanie Osób

Utwórz akcję Remove typu get.

W akcji utwórz parametr o nazwie `id` typu `int`, na podstawie, którego pobierz osobę z bazy i przekaż do widoku.

Utwórz widok dla akcji Remove. W widoku wyświetl dane osoby i przycisk potwierdzający usuwanie.

Przycisk potwierdzający usuwania, który znajduje się w widoku akcji Remove, musi zostać wysłany na akcję `RemoveConfirm` typu `post`.

Utwórz drugą akcję o nazwie `RemoveConfirm` typu `post`, która w parametrze przyjmuje parametr `id` typu `int`.

W akcji typu `post` usuń osobę z bazy na podstawie identyfikatora przekazanego w parametrze.

W widoku `Index`, przy każdym wpisie utwórz link kierujący na akcję Remove.



# Wyszukiwanie

W widoku Index utwórz formularz z polem na nazwisko osoby.

Formularz powinien zostać wysłany na akcję Index wraz z nazwiskiem osoby, którą chcemy znaleźć.

Zmień akcję Index, tak, aby przyjmowała dane formularza, przeszukała bazę i zwróciła w tym samym widoku znalezione osoby.

Powodzenia