|  |
| --- |
| ВЛАДИМИР ВАЛЕНТИНОВ РАДЕВ  1130 |
| **Обучаващ център: София-град**  **Водещ преподавател:** Ивайло Желев  **Група:** 1  **Училище:** НПМГ “Акад.Любомир Чакалов” |
|  |

# Проблем 1

|  |
| --- |
| Проектът не се стартира. Намерете решение за синтактичните и логически грешки, за да се стартира проекта успешно |
|  |

# Решение на проблем 1

|  |
| --- |
| **Трябва да се промени cONNECTION sTRINGA** С ТОЗИ НА ЛОКАЛНАТА МАШИНА НА КОЯТО СЕ РЪНВА ПРОЕКТА В data/MylibraryId/ optionsBuilder.UseSqlServer(TВОЯТ)**-В СЪЩИЯ ТОЗИ КЛАС ТРЯБВА ДА СЕ ПРОМЕНИ** public object books сpublic DbSet<Book> Books { get; set; }**-в boOKCONTROLLER.cs на ред 30,** трябва да е new BooksViewModel(), вместо new BOOK()**-съЩО ТАКА ТРЯБВА ПРОЕКТА ДА СЕ СМЕНИ НА WEB ЗА ДА РЪННЕ** |
|  |

# Проблем 2

|  |
| --- |
| **При създаване на нов запис за книга важат валидации, но при редактиране на запис същите липсват. Решете проблема като добавите валидации и при редакция** |
|  |

# Решение на проблем 2

|  |
| --- |
| В класа Booksеditviewmodel слагаме над пропертитата тези рекуарднати неща от BooksCreateViewModel.Пример преди поправката:BooksCreateViewModel: [Required]  [MaxLength(80, ErrorMessage = "Author name cannot be longer than 80 characters")] public string Author { get; set; }Booksеditviewmodel:public string Author { get; set; }Пример след поправката:BooksCreateViewModel: [Required]  [MaxLength(80, ErrorMessage = "Author name cannot be longer than 80 characters")] public string Author { get; set; }Booksеditviewmodel: [Required]  [MaxLength(80, ErrorMessage = "Author name cannot be longer than 80 characters")] public string Author { get; set; } |
|  |

# Проблем 3

|  |
| --- |
| При опит за редакция на съществуващ запис, вместо това се създава нов запис със редактираните данни |
|  |

# Решение на проблем 3

|  |
| --- |
| В bookController на метода public async Task<IActionResult> Edit(BooksEditViewModel model)  :когато създаваме нова книга трябва да сетнем ид-то й с това на подавания едитВиеу модел  **Преди:**  Book book = new Book  {  ISBN = model.ISBN,  Author = model.Author,  Pages = model.Pages,  Title = model.Title,  Price = model.Price  };  **След:**  Book book = new Book  {  Id = model.Id,  ISBN = model.ISBN,  Author = model.Author,  Pages = model.Pages,  Title = model.Title,  Price = model.Price  };  **Правим това за да можем да ъпдейтнем вече съществуваща книга с това Ид, а не да правим нова!**  **:**В същият клас метода  **Преди:**  private bool BookExists(int id)  {  return \_context.Books.Any(e => e.Id != id);  }  **Сега:**  private bool BookExists(int id)  {  return \_context.Books.Any(e => e.Id == id);  }  Правим това за да може не да проверя кои Идта не съвпадат, а кои съвпадат |
|  |

# Проблем 4

|  |
| --- |
| **В изгледа, показващ всички записи на книги има грешка при показването на записите – вместо цена се показва заглавието на книгата:** |
|  |

# Решение на проблем 4

|  |
| --- |
| В VIEW/BOOKS/INDEX.CSHTML NA 43 REDПРЕДИ: <td>  @Html.DisplayFor(modelItem => item.Title) </Td>**СЕГА:** <td>  @Html.DisplayFor(modelItem => item.Price) </Td>**НЕ СЕ ВЗИМА ITEM.PRICE A SE VZIMA ITEM.TITLE!** |
|  |