Министерство образования Республики Беларусь

Учреждение образования

Брестский государственный технический университет

Кафедра ИИТ

Лабораторная работа №2

За 5 семестр

По дисциплине «Веб технологии»

Тема: «Передача данных через параметры URL. Передача данных через HTML-форму. Основные принципы организации веб-приложения, использующего базы данных»

Выполнил: студент 3 курса

Группы ПО-4(2)

Коташевич С.Н.

Проверил: Михняев А. Л.

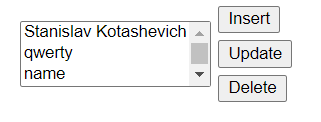
Брест 2021

Лабораторная работа №2

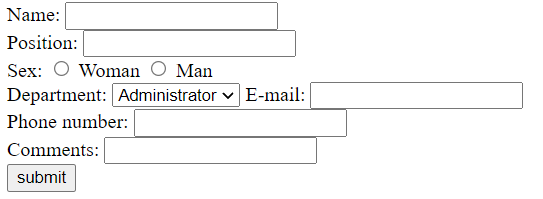
Цель: Освоить методы передачи данных в веб-среде. Разработать простейшее веб-приложение, использующее базы данных.

Ход работы:

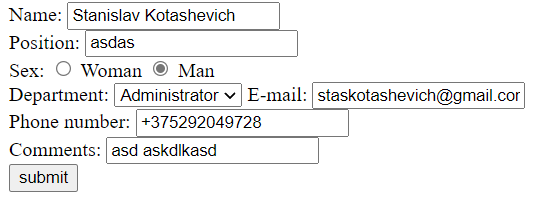
1. Создать базу данных, содержащую персональные данные о сотрудниках и отделах организации. Создать cfm – шаблон, выводящий актуальный список о сотрудниках организации. В интерфейсе предусмотреть кнопки ДОБАВИТЬ , ИЗМЕНТИЬ, УДАЛИТЬ.С помощью которых можно соответственно добавить данные о сотруднике в базу данных, изменить данные выбранного сотрудника и удалить данные выбранного сотрудника из базы данных.



Insert



Update



[HttpPost]

public IActionResult Third(string Action, string PersonId)

{

if (ModelState.IsValid)

{

return RedirectToAction(Action, new {PersonId});

}

return RedirectToAction("Index");

}

public IActionResult Insert()

{

return View();

}

public IActionResult Update(int personId)

{

return View(\_personContext.Persons.FirstOrDefault(p => p.Id == personId));

}

[HttpPost]

public IActionResult Update(Person person)

{

if (ModelState.IsValid)

{

var temp = \_personContext.Persons.FirstOrDefault(p => p.Id == person.Id);

if (temp != null)

{

temp.Name = person.Name;

temp.Comments = person.Comments;

temp.Department = person.Department;

temp.Email = person.Email;

temp.Phone = person.Phone;

temp.Position = person.Position;

temp.Sex = person.Sex;

\_personContext.SaveChanges();

}

}

return RedirectToAction("Index", \_personContext.Persons.ToList());

}

public IActionResult Delete(int personId)

{

if (ModelState.IsValid)

{

\_personContext.Remove(\_personContext.Persons.FirstOrDefault(p => p.Id == personId));

\_personContext.SaveChanges();

}

return RedirectToAction("Index", \_personContext.Persons.ToList());

}

[HttpPost]

public IActionResult Insert(Person person)

{

if (ModelState.IsValid)

{

var person1 = \_personContext.Persons.FirstOrDefault(p => p.Name == person.Name);

if (person1 == null)

{

\_personContext.Add(person);

\_personContext.SaveChanges();

}

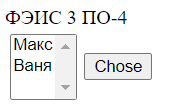
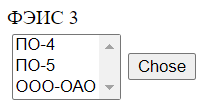
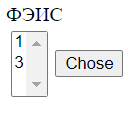
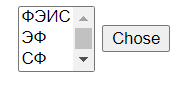
}

return RedirectToAction("Index", \_personContext.Persons.ToList());

}

}

1. С помощью уточняющих запросов отразить организационную структуру БГТУ: Список факультетов – Список Курсов – Список Групп – Список Студентов – Информация о выбранном студенте. Все данные для уточняющих запросов должна храниться в БД.



public IActionResult Index()

{

return View(\_dataBaseContext.Faculties.ToList());

}

[HttpGet]

public IActionResult Faculties(int FacultyId)

{

if (ModelState.IsValid)

{

return RedirectToAction("Courses", new {FacultyId});

}

return RedirectToAction("Index", \_dataBaseContext.Faculties.ToList());

}

public IActionResult Courses(int FacultyId)

{

var courses = \_dataBaseContext.Courses

.Include(f => f.Faculty)

.Where(c => c.FacultyId == FacultyId).Select(c => c).ToList();

return View(courses);

}

[HttpGet]

public IActionResult Groups(int FacultyId, int CourseId)

{

if (ModelState.IsValid)

{

return RedirectToAction("Group", new { FacultyId, CourseId });

}

return RedirectToAction("Index", \_dataBaseContext.Faculties.ToList());

}

public IActionResult Group(int FacultyId, int CourseId)

{

var courses = \_dataBaseContext.Groups

.Include(f => f.Course.Faculty)

.Include(c => c.Course)

.Where(c => c.CourseId == CourseId).Select(c => c).ToList();

return View(courses);

}

[HttpGet]

public IActionResult Students(int FacultyId, int CourseId, int GroupId)

{

if (ModelState.IsValid)

{

return RedirectToAction("Student", new { FacultyId, CourseId, GroupId });

}

return RedirectToAction("Index", \_dataBaseContext.Faculties.ToList());

}

public IActionResult Student(int FacultyId, int CourseId, int GroupId)

{

var courses = \_dataBaseContext.Students

.Include(f => f.Group.Course.Faculty)

.Include(c => c.Group.Course)

.Include(g => g.Group)

.Where(s => s.GroupId == GroupId)

.Select(s => s)

.ToList();

return View(courses);

}

[HttpGet]

public IActionResult StudentInfo(int FacultyId, int CourseId, int GroupId, int StudentId)

{

if (ModelState.IsValid)

{

return RedirectToAction("SelectedStudent", new { FacultyId, CourseId, GroupId, StudentId });

}

return RedirectToAction("Index", \_dataBaseContext.Faculties.ToList());

}

public IActionResult SelectedStudent(int FacultyId, int CourseId, int GroupId, int StudentId)

{

var student = \_dataBaseContext.Students

.Include(f => f.Group.Course.Faculty)

.Include(c => c.Group.Course)

.Include(g => g.Group)

.FirstOrDefault(s => s.Id == StudentId);

return View(student);

}

}

Вывод: освоил методы передачи данных в веб-среде. Разработал простейшее веб-приложение, использующее базы данных.