

**МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ**

**РОССИЙСКОЙ ФЕДЕРАЦИИ**

Брянский государственный технический университет

**Утверждаю**

**Ректор университета**

**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_О.Н.Федонин**

**«\_\_\_\_»\_\_\_\_\_\_\_\_\_\_\_\_2017 г.**

**Программирование в среде Microsoft .NET**

**Проектирование классов. Методы, конструкторы, свойства**

**Методические указания**

**к выполнению лабораторной работы №4**

**для студентов очной формы обучения по направлениям подготовки 02.03.03 – «Математическое обеспечение и администрирование информационных систем», 09.03.01 – «Информатика и вычислительная техника» и 09.03.04 – «Программная инженерия»**

**Брянск 2017**

УКД 004.43

Программирование в среде Microsoft .Net. Проектирование классов. Методы, конструкторы, свойства [Текст] + [Электронный ресурс]: Методические указания к выполнению лабораторной работы №4 для студентов очной формы обучения по направлениям подготовки 02.03.03 – «Математическое обеспечение и администрирование информационных систем», 09.03.01 – «Информатика и вычислительная техника» и 09.03.04 – «Программная инженерия». – Брянск: БГТУ, 2017. – 8 с.

Разработали:

Д.А.Коростелёв

канд. техн. наук, доц.

Рекомендовано кафедрой «Информатика и программное обеспечение» БГТУ (протокол №2 от 16.09.2016)

Научный редактор А.А.Азарченков

Редактор издательства Л.И.Афонина

Компьютерный набор Д.А.Коростелёв

Темплан 2017 г., п.273

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Подписано в печать 23.09.17. Формат 60х84 1/16 Бумага офсетная. Офсетная печать. Усл.печ.л. 1,8. Уч.-изд.л. 1,8 Тираж 1 экз. Заказ Бесплатно.

Издательство Брянского государственного технического университета

241035, Брянск, бульвар 50-летия Октября, 7, БГТУ. 58-82-49.

Лаборатория оперативной полиграфии БГТУ, ул. Институтская, 16.

**Цель работы**

Целью работы является изучение основ проектирования классов, описания их метод и свойств.

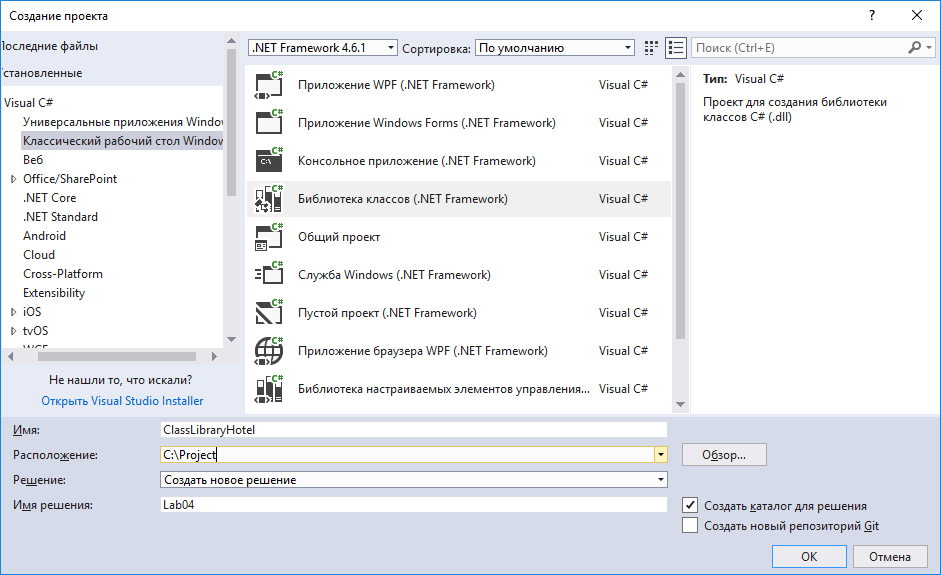
Продолжительность работы – 2 часа.

**Проектирование классов**

Рассмотрим особенности проектирования классов на примере организации работы гостиницы:

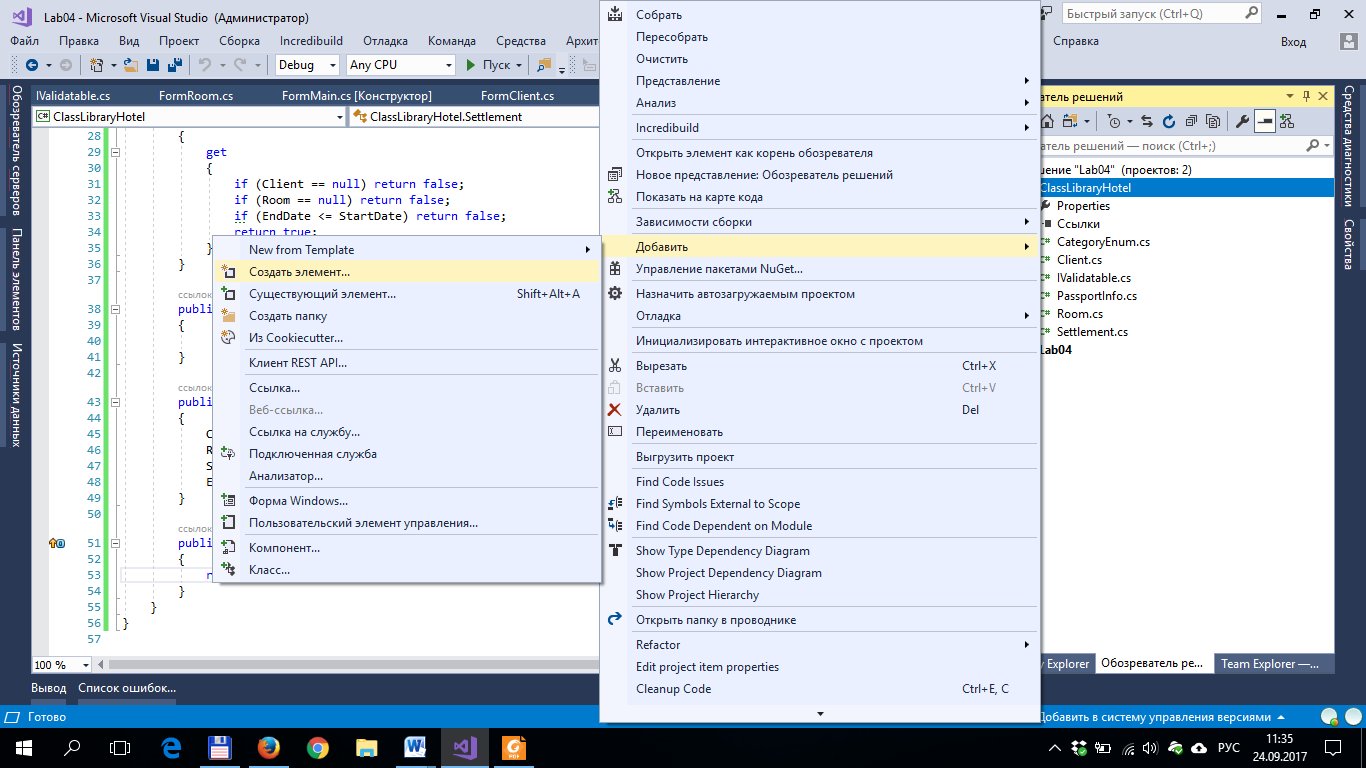
* класс «Клиент» (свойства «Фамилия», «Имя», «Отчество», «Паспортные данные»);
* класс «Номер» (свойства «Идентификатор», «Категория», «Стоимость в сутки», «Количество мест»);
* класс «Поселение» (свойства «Клиент», «Номер», «Дата начала проживания», «Дата окончания проживания»).

Для начала создадим новое решение на основе проекта типа «Библиотека классов» (Class Library) (рис. 1).

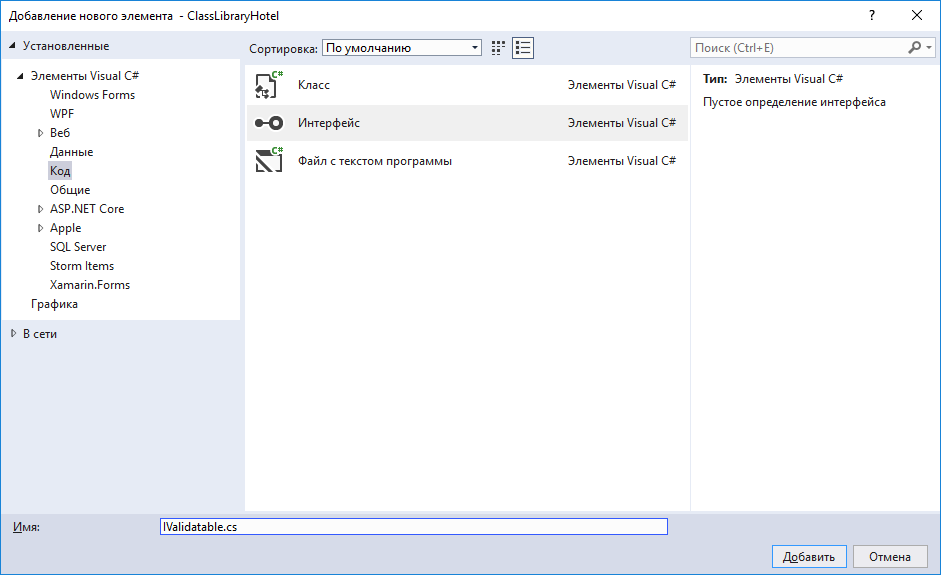


***Рис. 1. Создание нового решения и проекта «Библиотека классов»***

Добавим в проект новый файл для описания интерфейса, который будут реализовывать проектируемые классы (рис. 2-3).



***Рис. 2. Добавление нового элемента в проект***



***Рис. 3. Добавление интерфейса в проект***

В описание интерфейса добавим одно свойство – IsValid, которое будет информировать нас о том, что свойства объекта корректно заполнены:

namespace ClassLibraryHotel

{

interface IValidatable

{

bool IsValid { get; }

}

}

Т.к. «Клиент» содержит информацию о паспорте, а эта информация является структурированной, то для начала добавим и опишем класс «Паспортные данные» – **PassportInfo**:

using System;

namespace ClassLibraryHotel

{

public class PassportInfo

{

/// <summary>

/// Номер паспорта

/// </summary>

public string Number { get; set; } = "000000";

/// <summary>

/// Серия паспорта

/// </summary>

public string Seria { get; set; } = "0000";

/// <summary>

/// Дата выдачи паспорта

/// </summary>

public DateTime Date { get; set; } = DateTime.Today;

/// <summary>

/// Кем выдан паспорт

/// </summary>

public string Issuer { get; set; } = "";

public override string ToString()

{

return $"{Seria} {Number}";

}

}

}

Теперь добавим и опишем класс «Клиент» – **Client**:

namespace ClassLibraryHotel

{

/// <summary>

/// Клиент

/// </summary>

public class Client : IValidatable

{

/// <summary>

/// Имя

/// </summary>

public string FirstName { get; set; } = "";

/// <summary>

/// Отчество

/// </summary>

public string MiddleName { get; set; } = "";

/// <summary>

/// Фамилия

/// </summary>

public string LastName { get; set; } = "";

/// <summary>

/// Паспортные данные

/// </summary>

public PassportInfo Passport { get; set; } = new PassportInfo();

public bool IsValid

{

get

{

if (string.IsNullOrWhiteSpace(FirstName)) return false;

if (string.IsNullOrWhiteSpace(MiddleName)) return false;

if (string.IsNullOrWhiteSpace(LastName)) return false;

if (Passport == null) return false;

return true;

}

}

public Client()

{

}

public Client(string firstName, string lastName, string middleName)

{

FirstName = firstName;

LastName = lastName;

MiddleName = middleName;

}

public override string ToString()

{

return $"Фамилия: {LastName}\r\nИмя: {FirstName}\r\nОтчество: {MiddleName}\r\nПаспортные данные: {Passport}\r\n";

}

}

}

При описании гостиничного номера используется свойство «Категория». Т.к. у категории ограниченный набор возможных значений, то для описания этого свойства добавим перечисление – **CategoryEnum**:

namespace ClassLibraryHotel

{

/// <summary>

/// Категория номера

/// </summary>

public enum CategoryEnum

{

/// <summary>

/// Экономичный

/// </summary>

Econom,

/// <summary>

/// Стандартный

/// </summary>

Standart,

/// <summary>

/// Комфортный

/// </summary>

Comfort,

/// <summary>

/// Люкс

/// </summary>

Luxe

}

}

Описание класса **Room** будет следующим:

namespace ClassLibraryHotel

{

/// <summary>

/// Номер

/// </summary>

public class Room : IValidatable

{

/// <summary>

/// Идентификатор

/// </summary>

public int Number { get; set; } = 1;

/// <summary>

/// Категория

/// </summary>

public CategoryEnum Category { get; set; } = CategoryEnum.Econom;

/// <summary>

/// Стоимость в сутки

/// </summary>

public decimal PayByDay { get; set; } = 100;

/// <summary>

/// Количество мест

/// </summary>

public int PlaceCount { get; set; } = 2;

public bool IsValid

{

get

{

if (Number < 1) return false;

if (PayByDay <= 0) return false;

if (PlaceCount < 1) return false;

return true;

}

}

public Room()

{

}

public Room(int number, CategoryEnum category, decimal payByDay, int placeCount)

{

Number = number;

Category = category;

PayByDay = payByDay;

PlaceCount = placeCount;

}

public override string ToString()

{

return

$"Идентификатор: {Number}\r\nКатегория: {Category}\r\nСтоимость в сутки:{PayByDay}\r\nКоличество мест:{PlaceCount}\r\n";

}

}

}

Еще один класс «Поселение» – **Settlement** будем иметь следующее описание:

using System;

namespace ClassLibraryHotel

{

/// <summary>

/// Поселение

/// </summary>

public class Settlement : IValidatable

{

/// <summary>

/// Клиент

/// </summary>

public Client Client { get; set; }

/// <summary>

/// Номер

/// </summary>

public Room Room { get; set; }

/// <summary>

/// Дата начала проживания

/// </summary>

public DateTime StartDate { get; set; }

/// <summary>

/// Дата окончания проживания

/// </summary>

public DateTime EndDate { get; set; }

public bool IsValid

{

get

{

if (Client == null) return false;

if (Room == null) return false;

if (EndDate <= StartDate) return false;

return true;

}

}

public Settlement()

{

}

public Settlement(Client client, Room room, DateTime startDate, DateTime endDate)

{

Client = client;

Room = room;

StartDate = startDate;

EndDate = endDate;

}

public override string ToString()

{

return $"Клиент - {Client}\r\nНомер - {Room}\r\nПериод: {StartDate}-{EndDate}\r\n";

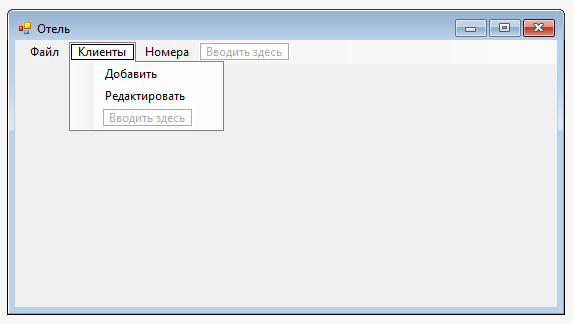
}

}

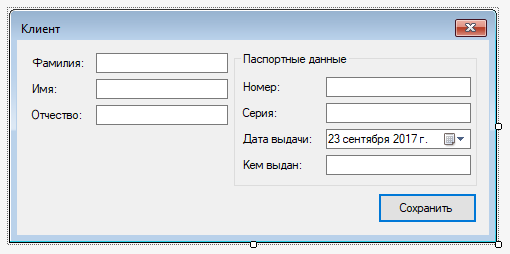
}

Для управления объектами описанных выше классов добавим в решение новый проект Windows Forms. В конструкторе формы добавим и настроим главное меню (рис. 4).

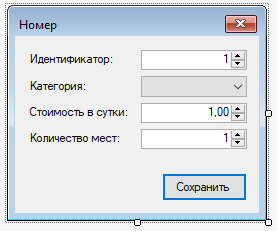
Создание и редактирование объектов классов «Клиент» и «Номер» реализуем через дополнительные диалоговые формы, которые будут вызываться при нажатии на пункты меню «Добавить» и «Редактировать» в соответствующих разделах меню «Клиенты» и «Номера». Интерфейс соответствующих диалоговых форм представлен на рис. 5-6.



***Рис. 4. Главное меню формы***



***Рис. 5. Диалоговая форма создания и редактирования клиента***



***Рис. 6. Диалоговая форма создания и редактирования номера***

Хранение клиента и номера реализуем через поля в классе главной формы:

private Client \_client;

private Room \_room;

Для пунктов меню добавим обработчики, которые будут вызывать соответствующие диалоговые формы:

private void addClientToolStripMenuItem\_Click(object sender, EventArgs e)

{

\_client = new Client();

FormClient formClient = new FormClient(\_client);

if (formClient.ShowDialog() == DialogResult.OK)

{

\_client = formClient.Client;

}

}

private void editClientToolStripMenuItem\_Click(object sender, EventArgs e)

{

FormClient formClient = new FormClient(\_client);

if (formClient.ShowDialog() == DialogResult.OK)

{

\_client = formClient.Client;

}

}

private void addRoomToolStripMenuItem\_Click(object sender, EventArgs e)

{

\_room = new Room();

FormRoom formRoom = new FormRoom(\_room);

if (formRoom.ShowDialog() == DialogResult.OK)

{

\_room = formRoom.Room;

}

}

private void editRoomToolStripMenuItem\_Click(object sender, EventArgs e)

{

FormRoom formRoom = new FormRoom(\_room);

if (formRoom.ShowDialog() == DialogResult.OK)

{

\_room = formRoom.Room;

}

}

Чтобы обеспечить передачу данных между диалоговой формой и главной формой дополним описание конструктора формы, а также реализуем обработчик нажатия кнопки «Сохранить». Для формы редактирования клиента исходный код будет выглядеть следующим образом:

using System;

using System.Windows.Forms;

using ClassLibraryHotel;

namespace Lab04

{

public partial class FormClient : Form

{

public Client Client { get; }

public FormClient(Client client)

{

InitializeComponent();

Client = client;

textBoxFirstName.Text = Client.FirstName;

textBoxMiddleName.Text = Client.MiddleName;

textBoxLastName.Text = Client.LastName;

textBoxIssuer.Text = Client.Passport.Issuer;

textBoxSeria.Text = Client.Passport.Seria;

textBoxNumber.Text = Client.Passport.Number;

dateTimePickerDate.Value = Client.Passport.Date;

}

private void buttonSave\_Click(object sender, EventArgs e)

{

Client.FirstName = textBoxFirstName.Text;

Client.MiddleName = textBoxMiddleName.Text;

Client.LastName = textBoxLastName.Text;

Client.Passport.Issuer = textBoxIssuer.Text;

Client.Passport.Seria = textBoxSeria.Text;

Client.Passport.Number = textBoxNumber.Text;

Client.Passport.Date = dateTimePickerDate.Value;

}

}

}

Исходный код конструктора формы и обработчика нажатия кнопки «Сохранить» для формы редактирования номера будет выглядеть следующим образом:

using System;

using System.Windows.Forms;

using ClassLibraryHotel;

namespace Lab04

{

public partial class FormRoom : Form

{

public Room Room { get; }

public FormRoom(Room room)

{

InitializeComponent();

Room = room;

comboBoxCategory.Items.Add(CategoryEnum.Econom);

comboBoxCategory.Items.Add(CategoryEnum.Standart);

comboBoxCategory.Items.Add(CategoryEnum.Comfort);

comboBoxCategory.Items.Add(CategoryEnum.Luxe);

comboBoxCategory.SelectedItem = Room.Category;

numericUpDownNumber.Value = Room.Number;

numericUpDownPayByDay.Value = Room.PayByDay;

numericUpDownPlaceCount.Value = Room.PlaceCount;

}

private void buttonSave\_Click(object sender, EventArgs e)

{

Room.Category = (CategoryEnum)comboBoxCategory.SelectedItem;

Room.Number = (int)numericUpDownNumber.Value;

Room.PayByDay = numericUpDownPayByDay.Value;

Room.PlaceCount = (int)numericUpDownPlaceCount.Value;

}

}

}

**Задание на лабораторную работу**

1. Создайте новый проект типа «Библиотека классов» (Class Library).
2. В проект библиотеки классов добавьте интерфейс IValidatable, описывающий метод свойство IsValid.
3. Разработайте 3 класса для предметной области согласно варианту (табл. 1). При необходимости разработайте для описания свойств классов вспомогательные классы или перечисления. Классы должны реализовывать интерфейс IValidatable, а также должны содержать конструкторы по умолчанию и с параметрами. Доступ к членам-данным в классе должен осуществляться извне толь через свойства.
4. Для каждого разработанного класса переопределите метод ToString().
5. Добавьте пояснения в виде xml-комментариев для классов, методов и их свойств.
6. Добавьте в решение новое приложение типа «Windows Forms». На форму добавьте и настройте главное меню.
7. В проект «Windows Forms» добавьте две диалоговые формы для создания и редактирования объектов первых двух классов соответственно. При разработке форм необходимо давать элементам управления соответствующие свойствам названия. Диалоговые формы должны иметь кнопку сохранения. Используемые элементы управления должны быть корректно настроены для соответствующих свойств классов (иметь правильный тип и допустимый диапазон значений).
8. Добавьте обработчики для пунктов меню, вызывающие формы создания и редактирования объектов.

Таблица 1

Варианты заданий

|  |  |
| --- | --- |
| Вариант | Классы |
| 1 | **Страховая компания**   1. Вид страхования (наименование). 2. Филиал (наименование филиала, адрес, телефон). 3. Договоры (номер договора, дата заключения, страховая сумма, тарифная ставка, филиал, вид страхования). |
| 2 | **Ломбард**   1. Клиент (фамилия, имя, отчество, номер паспорта, серия паспорта, дата выдачи паспорта). 2. Категории товаров (название, примечание). 3. Сдача в ломбард (категория товаров, клиент, описание товара, дата сдачи, дата возврата, сумма, комиссионные). |

Продолжение таблицы 1

|  |  |
| --- | --- |
| Вариант | Классы |
| 3 | **Реализация готовой продукции**   1. Товары (наименование, оптовая цена, розничная цена, описание). 2. Покупатели (телефон, контактное лицо, адрес). 3. Сделки (дата сделки, товар, количество, покупатель, признак оптовой продажи). |
| 4 | **Ведение заказов**   1. Заказчики (наименование, адрес, телефон, контактное лицо). 2. Товары (цена, доставка, описание). 3. Заказы (заказчик, товар, количество, дата). |
| 5 | **Бюро по трудоустройству**   1. Работодатели (название, вид деятельности, адрес, телефон). 2. Соискатели (фамилия, имя, отчество, квалификация, вид деятельности, иные данные, предполагаемый размер заработной платы). 3. Сделки (работодатель, соискатель, должность, комиссионные). |
| 6 | **Нотариальная контора**   1. Клиенты (название, вид деятельности, адрес, телефон). 2. Услуги (название, описание). 3. Сделки (клиент, услуга, сумма, комиссионные, описание). |
| 7 | **Фирма по продаже запчастей**   1. Поставщики (название, адрес, телефон). 2. Детали (название, артикул, цена, примечание). 3. Поставки (поставщик, деталь, количество, дата). |
| 8 | **Курсы по повышению квалификации**   1. Группы (специальность, отделение, количество студентов). 2. Преподаватели (фамилия, имя, отчество, телефон, стаж). 3. Нагрузка (преподаватель, группа, количество часов, предмет, тип занятия, оплата). |
| 9 | **Определение факультативов для студентов**   1. Студенты (фамилия, имя, отчество, адрес, телефон). 2. Предметы (название, объем лекций, объем практик, объем лабораторных работ). 3. Учебный план (студент, предмет, оценка). |
| 10 | **Распределение учебной нагрузки**   1. Преподаватели (фамилия, имя, отчество, ученая степень, должность, стаж). 2. Предметы (название, количество часов). 3. Нагрузка (преподаватель, предмет, номер группы). |
| 11 | **Распределение дополнительных обязанностей**   1. Сотрудники (фамилия, имя, отчество, оклад). 2. Виды работ (описание, оплата за день). 3. Работы (сотрудник, вид работ, дата начала, дата окончания). |

Продолжение таблицы 1

|  |  |
| --- | --- |
| Вариант | Классы |
| 12 | **Техническое обслуживание станков**   1. Виды станков (страна, год выпуска, марка). 2. Виды ремонта (название, продолжительность, стоимость, примечания). 3. Ремонт (вид станка, ремонт, дата начала, примечания). |
| 13 | **Туристическая фирма**   1. Маршруты (страна, климат, длительность, отель, стоимость). 2. Клиенты (фамилия, имя, отчество, адрес, телефон). 3. Путевки (маршрут, клиент, дата отправления, количество, скидка). |
| 14 | **Грузовые перевозки**   1. Маршруты (название, дальность, количество дней в пути, оплата). 2. Водители (фамилия, имя, отчество, стаж). 3. Проделанная работа (маршрут, водитель, дата отправки, дата возвращения, премия). |
| 15 | **Учет телефонных переговоров**   1. Абоненты (номер телефона, ИНН, адрес). 2. Города (название, тариф дневной, тариф ночной). 3. Переговоры (абонент, город, дата, количество минут, время суток). |
| 16 | **Учет внутриофисных расходов**   1. Отделы (название, количество сотрудников). 2. Виды расходов (название, описание, предельная норма). 3. Расходы (вид расходов, отдел, сумма, дата). |
| 17 | 1. Книги (название, автор, залоговая стоимость, стоимость проката, жанр). 2. Читатели (фамилия, имя, отчество, адрес, телефон). 3. Выданные книги (книга, читатель, дата выдачи, дата возврата). |
| 18 | **Прокат автомобилей**   1. Автомобили (марка, стоимость, стоимость проката, тип). 2. Клиенты (фамилия, имя, отчество, адрес, телефон). 3. Выданные автомобили (автомобиль, клиент, дата выдачи, дата возврата). |
| 19 | **Выдача банком кредитов**   1. Виды кредитов (название, условия получения, ставка, срок). 2. Клиенты (название, вид собственности, адрес, телефон, контактное лицо). 3. Кредиты (вид кредитов, клиент, сумма, дата выдачи). |

Окончание таблицы 1

|  |  |
| --- | --- |
| Вариант | Классы |
| 20 | **Инвестирование свободных средств**   1. Ценные бумаги (код ценной бумаги, минимальная сумма сделки, рейтинг, доходность за прошлый год, дополнительная информация). 2. Клиенты (клиент, название, вид собственности, адрес, телефон). 3. Инвестиции (ценная бумага, клиент, котировка, дата покупки, дата продажи). |

**Контрольные вопросы**

1. Каким образом объявляются и вызываются методы?
2. Каким образом осуществляется перегрузка методов?
3. Что такое свойства?
4. Каким образом объявляются конструкторы на языке?
5. Что такое частичные классы?
6. Каким образом реализовано множественное наследование в среде Microsoft .Net?
7. Какие особенности наследования интерфейсов?

# Список рекомендуемой литературы

1. Нейгел, К. C# 5.0 и платформа .NET 4.5 для профессионалов / К.Нейгел, Б.Ивьен, Дж.Глинн, М.Скиннер, К.Уотсон. – М.: Диалектика, 2013. – 1440 с.
2. Неш, Т. C# 2010: ускоренный курс для профессионалов / Т.Неш – М.: Вильямс, 2010. – 592 с.
3. Снелл, М. Microsoft Visual Studio 2008 / М.Снелл, Л.Пауэрс. – СПб: БХВ-Петербург, 2009. – 1200 с.
4. Троелсен, Э. Язык программирования С# 5.0 и платформа .NET 4.5 / Э.Троелсен. – 6-е изд. – М.: Вильямс, 2015. – 1312 с.
5. Рихтер, Дж. CLR via C#. Программирование на платформе Microsoft .NET Framework 4.5 на языке C# / Дж.Рихтер. – СПб.: Питер, 2016. – 896 с.