

МИНОБРНАУКИ РОССИИ  
Бугултминский филиал федерального государственного бюджетного  
образовательного учреждения высшего образования  
«Казанский национальный исследовательский  
технологический университет»  
(БФ ФГБОУ ВО «КНИТУ»)

БФ ФГБОУ ВО «КНИТУ»

КОНТРОЛЬНАЯ РАБОТА  
по дисциплине (модулю)  
ИНФОРМАЦИОННЫЕ ТЕХНОЛОГИИ  
(*наименование дисциплины (модуля)*)  
09.03.02 Информационные системы и технологии  
(*код и наименование направления подготовки/ специальности*)  
Информационные системы и технологии  
(*наименование профиля/программы/направленности/специализации*)  
бакалавр  
*квалификация*

Бугульма, 2025

## **Задание 1. Теоретические основы информационных технологий**

**Цель задания:** Проверка теоретических знаний об основных понятиях, процессах и классификациях информационных технологий, а также понимания перспективных направлений развития ИТ (искусственный интеллект, мультимедиа).

### **Проверяемые знания и умения:**

- Знание определений информационной технологии (ИТ) и информационного процесса.
- Умение классифицировать ИТ по различным признакам.
- Понимание этапов работы с информацией: от сбора до защиты.
- Способность приводить примеры применения прикладных ИТ в различных профессиональных сферах.

### **Задания по вариантам**

Студенту необходимо подготовить краткий письменный ответ (объемом 1 – 1,5 страницы) по одной из следующих тем:

1. Информационная технология: определение, цели и основные компоненты.
2. Базовые информационные процессы: сбор и подготовка информации.
3. Процесс хранения информации: методы, технические средства и эволюция носителей.
4. Обработка информации как центральный этап ИТ: виды и примеры.
5. Передача информации: каналы связи, протоколы и сетевые технологии.
6. Информационная безопасность: основные угрозы и методы защиты данных.
7. Классификация ИТ по типу обрабатываемой информации (данные, текст, графика, знания).
8. Классификация ИТ по типу пользовательского интерфейса (командный, графический, речевой).
9. Базовые и прикладные информационные технологии: отличия и примеры взаимодействия.
10. Мультимедиа-технологии: понятие, компоненты и сферы применения.
11. Гипертекстовые технологии: принципы организации данных и роль в World Wide Web.
12. Технологии офисной автоматизации: задачи, инструменты и влияние на производительность.
13. Искусственный интеллект (ИИ): определение и основные области применения сегодня.
14. Экспертные системы: структура, назначение и примеры использования в медицине или технике.
15. Системы поддержки принятия решений (СППР): роль в управлении организацией.
16. Геоинформационные технологии (ГИС): назначение и примеры использования.

17. Технологии «клиент-сервер»: понятие и преимущества для распределенной обработки данных.
1. Облачные технологии: модели обслуживания (*SaaS*, *PaaS*, *IaaS*) и их преимущества.
2. Понятие автоматизированного рабочего места (АРМ): состав и назначение.
3. Эволюция информационных технологий: от «бумажных» до глобальных компьютерных сетей.

### Методические рекомендации:

1. **Раскрытие понятия:** Начните с четкого определения термина. Например, ИТ — это совокупность методов и инструментов для работы с данными.
2. **Структура ответа:** Используйте логическую последовательность: введение (определение), основная часть (характеристики или классификация), примеры и заключение.
3. **Классификация:** Приводя классификацию, обязательно указывайте признак, по которому она проводится (например, «по типу интерфейса» или «по области применения»).
4. **Примеры:** Каждое теоретическое положение подкрепляйте конкретным примером из жизни или будущей профессиональной деятельности (например, использование ИИ для распознавания лиц).
5. **Оформление:** Ответ должен быть структурированным, с использованием заголовков и списков для наглядности.

### Образец решения. Вариант 0.

#### Тема: Процесс обработки информации как этап ИТ.

**1. Определение:** Обработка информации — это последовательность действий, выполняемых над данными для получения результата, пригодного для принятия решений или дальнейшего использования. Это «сердце» любой информационной технологии.

#### 2. Виды обработки:

- **Арифметическая и логическая:** выполнение вычислений и сравнений.
- **Сортировка и группировка:** упорядочивание данных по определенному признаку (например, по алфавиту).
- **Агрегирование:** создание сводных отчетов (подсчет сумм, средних значений).

**3. Инструменты:** Основным инструментом обработки сегодня является центральный процессор (*CPU*) компьютера и специализированное программное обеспечение (текстовые редакторы, электронные таблицы, системы управления базами данных).

**4. Пример:** В банковской системе процесс обработки происходит, когда клиент переводит деньги. Система проверяет баланс (логическая операция), вычитает сумму перевода (арифметическая операция) и записывает новый остаток в базу данных.

**5. Заключение:** Без этапа обработки данные остаются «сырыми» и не представляют ценности для пользователя. Качественная обработка позволяет превратить хаотичный набор цифр в полезную информацию.

## Задание 2. Структурное моделирование информационной системы в IDEF0

**Цель задания:** Освоить методологию функционального моделирования *IDEF0*, научиться выделять границы системы, определять управляющие воздействия, ресурсы и выполнять системную декомпозицию сложных процессов.

### Проверяемые знания и умения:

- Знание синтаксиса нотации *IDEF0* (входы, выходы, управление, механизмы).
- Умение формулировать основную функцию системы (контекст).
- Навыки декомпозиции процесса на функциональные блоки.
- Умение соблюдать правила доминирования и связности дуг.

### Задания по вариантам

Для каждой предметной области необходимо: построить контекстную диаграмму  $A - 0$  и провести декомпозицию первого уровня (диаграмма  $A0$ ), выделив 3 – 5 блоков.

1. **Библиотека:** процесс обслуживания читателей и учета книжного фонда.
2. **Школьный журнал:** процесс ведения учета успеваемости и посещаемости.
3. **Интернет-магазин:** процесс обработки и доставки заказов покупателям.
4. **Поликлиника:** процесс записи на прием и оказания медицинской помощи.
5. **Складской учёт:** процесс приемки, хранения и отгрузки товаров.
6. **Гостиница:** процесс бронирования номеров и размещения гостей.
7. **Автосервис:** процесс приема заявок, ремонта и выдачи автомобилей.
8. **Учебный центр:** процесс организации курсов и обучения слушателей.
9. **Доставка заказов (курьерская служба):** процесс логистики от склада до двери.
10. **Фитнес-клуб:** процесс продажи абонементов и проведения тренировок.
11. **Аптека:** процесс закупки лекарств и розничной торговли.
12. **Кадровое агентство:** процесс подбора персонала для компаний-заказчиков.
13. **Стоматологическая клиника:** процесс лечения и протезирования зубов.
14. **Туристическая фирма:** процесс формирования и реализации турпакетов.
15. **Производство мебели:** процесс от приема эскиза до сборки изделия.
16. **Автозаправочная станция (АЗС):** процесс реализации топлива и сопутствующих товаров.
17. **Ветеринарная клиника:** процесс обследования и лечения животных.
18. **Клининговая компания:** процесс приема заявок и оказания услуг по уборке.
19. **Кинотеатр:** процесс продажи билетов и организации кинопоказов.
20. **Пиццерия:** процесс приема заказов, приготовления и продажи пиццы.

## Методические рекомендации:

1. **Контекстная диаграмма (А-0):** Это «вид сверху». Здесь всегда только один блок, представляющий всю систему целиком.
  - **Вход (*Input* - слева):** Объекты или данные, которые преобразуются (например, «Заявка клиента»).
  - **Управление (*Control* - сверху):** Правила, законы, регламенты (например, «Устав фирмы», «Законодательство РФ»).
  - **Механизмы (*Mechanism* - снизу):** Кто или что выполняет работу (например, «Персонал», «Компьютерная программа»).
  - **Выход (*Output* - справа):** Результат работы (например, «Оказанная услуга»).
2. **Декомпозиция (А0):** Разбейте основной блок на несколько последовательных этапов. Помните о «законе сохранения дуг»: все стрелки с контекстной диаграммы должны появиться на диаграмме декомпозиции.
3. **Имена:** Блоки называйте глаголами или глагольными оборотами («Оформить заказ»), а стрелки — существительными («Договор»).

## Образец решения. Вариант 0: Предметная область: Деятельность кофейни.

### 1. Контекстная диаграмма А-0:

- **Центральный блок:** «Обеспечить работу кофейни».
- **Входы (*I*):** Ингредиенты (кофе, молоко, сахар), клиенты.
- **Управления (*C*):** Рецептуры напитков, СанПиН, правила внутреннего распорядка.
- **Механизмы (*M*):** Бариста, кофемашина, кассовый аппарат.
- **Выходы (*O*):** Готовый напиток, чек, довольный клиент.

### 2. Декомпозиция А0 (основные процессы):

4. **Прием и оплата заказа:** Вход — Клиент; Выход — Чек и информация о заказе.
5. **Приготовление напитка:** Вход — Ингредиенты; Управление — Рецептура; Механизм — Бариста и оборудование.
6. **Выдача заказа:** Вход — Готовый напиток; Выход — Клиент с напитком.

### **Задание 3. Моделирование потоков данных DFD и описание процессов IDEF3**

**Цель задания:** Освоить методы описания документооборота и логики выполнения работ. Научиться разграничивать потоки данных (нотация *DFD*) и последовательность технологических операций (нотация *IDEF3*).

#### **Проверяемые знания и умения:**

- Знание элементов диаграммы потоков данных (*DFD*): внешние сущности, процессы, хранилища данных, потоки.
- Умение моделировать передачу информации между системой и внешним миром.
- Навыки логического описания сценариев выполнения работ с использованием перекрестков (AND, OR, XOR) в нотации *IDEF3*.

#### **Задания по вариантам**

Для каждой предметной области необходимо: выделить внешние сущности и хранилища данных, построить *DFD*-диаграмму (текстовое описание или схема) и описать один сложный процесс в логике *IDEF3*.

1. **Библиотека:** процесс регистрации нового читателя и выдачи книги.
2. **Школьный журнал:** процесс выставления итоговых оценок за четверть.
3. **Интернет-магазин:** процесс возврата товара ненадлежащего качества.
4. **Поликлиника:** процесс формирования медицинской карты и записи на обследование.
5. **Складской учёт:** процесс проведения инвентаризации остатков.
6. **Гостиница:** процесс онлайн-бронирования через агрегатор.
7. **Автосервис:** процесс закупки запчастей под конкретный заказ.
8. **Учебный центр:** процесс формирования учебных групп и расписания.
9. **Доставка заказов:** процесс распределения заказов между курьерами.
10. **Фитнес-клуб:** процесс записи на персональную тренировку.
11. **Аптека:** процесс проверки рецепта и продажи лекарственного препарата.
12. **Кадровое агентство:** процесс прохождения собеседования кандидатом.
13. **Стоматологическая клиника:** процесс составления плана лечения.
14. **Туристическая фирма:** процесс оформления визы для клиента.
15. **Производство мебели:** процесс закупки материалов у поставщиков.
16. **Автозаправочная станция (АЗС):** процесс заправки по топливным картам.
17. **Ветеринарная клиника:** процесс вакцинации питомца.
18. **Клининговая компания:** процесс оценки стоимости работ на объекте.
19. **Кинотеатр:** процесс возврата электронного билета.
20. **Пиццерия:** процесс формирования отчета о продажах за смену.

## Методические рекомендации:

1. **Отличие DFD от IDEF0:** Если в *IDEF0* мы смотрим на функции («Что делается?»), то в *DFD* мы смотрим на данные («Что передается?»). В *DFD* нет «Управления» и «Механизмов», зато есть **Хранилища данных** (базы данных, папки с документами).
2. **Внешние сущности:** Это объекты вне системы (Клиент, Банк, Налоговая), которые являются источниками или приемниками информации.
3. **Хранилища данных:** Места, где информация лежит «в покое» (Таблица «Товары», Файл «Сотрудники»).
4. **IDEF3:** Используется для описания логики. Используйте перекрестки:
  - **AND (&):** выполняются все последующие действия.
  - **XOR (X):** выполняется строго одно из нескольких действий.
  - **OR (O):** выполняется одно или несколько действий.

## Образец решения. Вариант 0: Предметная область: Прокат велосипедов.

### 1. Моделирование DFD:

- **Внешние сущности:** Клиент, Администратор.
- **Процессы:** 1. Регистрация клиента. 2. Оформление аренды. 3. Прием оплаты.
- **Хранилища данных:** БД «Клиенты», БД «Велосипеды», БД «Транзакции».
- **Потоки:** «Данные паспорта» (от Клиента в Процесс 1), «Информация о наличии» (из БД «Велосипеды» в Процесс 2).

### 2. Описание процесса IDEF3 (Процесс «Выдача велосипеда»):

5. Осмотр велосипеда клиентом.
6. **Перекресток XOR:**
  - А) Клиента устраивает состояние → Переход к шагу 3.
  - Б) Клиента не устраивает состояние → Выбор другого велосипеда (возврат к шагу 1).
7. Подписание договора.
8. Внесение залога.
9. **Перекресток AND:**
  - А) Выдача ключа от замка.
  - Б) Отметка в БД о начале аренды.

## Задание 4. Проектирование ER-модели базы данных

**Цель задания:** Научиться проектировать логическую структуру базы данных, выделять сущности предметной области, определять их атрибуты и устанавливать смысловые связи между ними, обеспечивая целостность данных.

### Проверяемые знания и умения:

- Умение отличать сущность (объект) от атрибута (свойства).
- Навыки определения первичных ключей (*PK*).
- Умение устанавливать типы связей ( $1: 1$ ,  $1: M$ ,  $M: N$ ).
- Навык нормализации данных (преобразование связей «многие-ко-многим»).

### Задания по вариантам

Для каждой предметной области необходимо выделить не менее 4 сущностей, указать их атрибуты, первичные ключи и связи.

1. **Библиотека:** учет книг, авторов, читателей и фактов выдачи литературы.
2. **Школьный журнал:** ученики, учителя, предметы и оценки в журнале.
3. **Интернет-магазин:** товары, категории, клиенты и заказы.
4. **Поликлиника:** врачи, пациенты, кабинеты и записи на прием.
5. **Складской учёт:** товары, поставщики, склады и накладные прихода/расхода.
6. **Гостиница:** номера, категории номеров, гости и регистрации (заселения).
7. **Автосервис:** клиенты, автомобили, мастера и наряды на выполненные работы.
8. **Учебный центр:** курсы, преподаватели, студенты и ведомости об окончании.
9. **Доставка заказов:** курьеры, заказы, клиенты и точки выдачи.
10. **Фитнес-клуб:** тренеры, клиенты, типы абонементов и посещения.
11. **Аптека:** лекарства, производители, продажи и рецепты.
12. **Кадровое агентство:** вакансии, компании-работодатели, соискатели и отклики.
13. **Стоматологическая клиника:** пациенты, стоматологи, услуги и счета за лечение.
14. **Туристическая фирма:** туры, страны, отели и проданные путевки.
15. **Производство мебели:** изделия, материалы, цеха и заказы покупателей.
16. **Автозаправочная станция (АЗС):** виды топлива, колонки, операторы и чеки.
17. **Ветеринарная клиника:** владельцы, питомцы, виды процедур и история болезни.
18. **Клининговая компания:** услуги, сотрудники, заказы и инвентарь.
19. **Кинотеатр:** фильмы, залы, сеансы и билеты.
20. **Пиццерия:** ингредиенты, рецепты пиццы, заказы и повара.

### Методические рекомендации:

1. **Выделение сущностей:** Сущность — это класс однотипных объектов (например, «Студент»), а не конкретный человек. Сущность обычно является существительным.
2. **Атрибуты:** Это свойства сущности (например, у «Студента» есть «ФИО», «Номер зачетки»).
3. **Первичный ключ (PK):** Это уникальный идентификатор записи (например, ID\_Студента). Он не может повторяться.
4. **Типы связей:**
  - **1: M (Один-ко-многим):** У одного отдела много сотрудников. (Самый частый тип).
  - **M: N (Многие-ко-многим):** Один студент посещает много курсов, на один курс ходит много студентов.
5. **Разрешение связи M: N:** В реляционных БД такая связь реализуется через третью (промежуточную) таблицу, которая содержит ключи обеих связываемых таблиц.

### Образец решения. Вариант 0: Предметная область: Ведение учета в автошколе.

Сущность	Атрибуты	Тип ключа
<b>Инструктор</b>	ID_Инструктора, ФИО, Стаж, Автомобиль	<i>PK</i>
<b>Курсант</b>	ID_Курсанта, ФИО, Телефон, Номер договора	<i>PK</i>
<b>Занятие</b>	ID_Занятия, Дата, Время, Оценка	<i>PK</i>
<b>Учебная_Группа</b>	ID_Группы, Название, Категория (B, C)	<i>PK</i>

### Связи:

1. **Учебная\_Группа — Курсант:** Связь 1: M (В одной группе много курсантов, курсант только в одной группе).
2. **Инструктор — Занятие:** Связь 1: M (Один инструктор проводит много занятий).
3. **Курсант — Занятие:** Связь 1: M (Один курсант посещает много занятий).
4. **Инструктор — Учебная\_Группа:** Связь M: N. *Решение:* Создаем промежуточную таблицу «Назначение\_инструкторов» (ключи группы и инструктора). Один инструктор может вести теорию в разных группах, в группе могут быть разные инструкторы (теория/практика).

## Задание 5. Проектирование простой информационной системы с использованием Entity Framework

**Цель задания:** Научиться проектировать архитектуру приложения на основе объектно-реляционного отображения (ORM). Освоить принципы создания моделей данных, настройки связей между ними и написания запросов к данным с использованием технологии *EntityFramework* и языка *LINQ*.

### Проверяемые знания и умения:

- Умение проектировать классы-модели данных.
- Понимание принципов работы *EntityFramework* (подход *Code First*).
- Навыки описания CRUD-операций (*Create, Read, Update, Delete*).
- Умение составлять *LINQ*-запросы для фильтрации, сортировки и агрегации данных.

### Задания по вариантам

Для каждой предметной области необходимо описать структуру классов, связи и привести примеры программного взаимодействия с данными.

1. **Библиотека:** модели «Книга» и «Автор», запрос на поиск всех книг конкретного автора.
2. **Школьный журнал:** модели «Ученик» и «Оценка», расчет среднего балла ученика.
3. **Интернет-магазин:** модели «Товар» и «Категория», фильтрация товаров по цене.
4. **Поликлиника:** модели «Врач» и «Прием», группировка приемов по врачам.
5. **Складской учёт:** модели «Товар» и «Склад», получение списка товаров с остатком меньше 5 единиц.
6. **Гостиница:** модели «Номер» и «Бронирование», поиск свободных номеров определенной категории.
7. **Автосервис:** модели «Автомобиль» и «Заказ-наряд», сортировка заказов по дате.
8. **Учебный центр:** модели «Курс» и «Студент», подсчет количества студентов на курсе.
9. **Доставка заказов:** модели «Курьер» и «Доставка», выборка всех доставок конкретного курьера за день.
10. **Фитнес-клуб:** модели «Клиент» и «Абонемент», поиск клиентов, чей абонемент истекает.
11. **Аптека:** модели «Лекарство» и «Продажа», расчет общей выручки за смену.
12. **Кадровое агентство:** модели «Вакансия» и «Отклик», фильтрация вакансий по зарплате.
13. **Стоматологическая клиника:** модели «Пациент» и «Процедура», история посещений пациента.
14. **Туристическая фирма:** модели «Тур» и «Страна», поиск самых дешевых туров в выбранную страну.

15. **Производство мебели:** модели «Изделие» и «Заказ», статус выполнения заказа.
16. **Автозаправочная станция (АЗС):** модели «Колонка» и «Транзакция», отчет по проданному топливу.
17. **Ветеринарная клиника:** модели «Питомец» и «Прием», поиск приемов для конкретного вида животного.
18. **Клининговая компания:** модели «Сотрудник» и «Заявка», распределение заявок.
19. **Кинотеатр:** модели «Сеанс» и «Фильм», выборка сеансов на конкретную дату.
20. **Пиццерия:** модели «Пицца» и «Заказ», расчет стоимости заказа.

### Методические рекомендации:

1. **Определение моделей:** Создайте классы, где свойства соответствуют полям таблицы. Используйте Id как первичный ключ.
2. **Связи:** Для реализации связи «один-ко-многим» добавьте в модель «много» свойство внешнего ключа (например, CategoryId) и навигационное свойство (например, public virtual Category Category { get; set; }).
3. **CRUD-операции:**
  - *Create:* db.Items.Add(newItem); db.SaveChanges();
  - *Read:* db.Items.ToList();
  - *Update:* Изменение свойств объекта и db.SaveChanges();
  - *Delete:* db.Items.Remove(item); db.SaveChanges();
4. **LINQ-запросы:** Используйте методы .Where() для фильтрации, .OrderBy() для сортировки и .GroupBy() для группировки.

### Образец решения. Вариант 0: Предметная область: Учет заказов в кофейне.

#### 1. Модели данных:

```
public class Drink {
    public int Id { get; set; }
    public string Name { get; set; } // Название напитка
    public decimal Price { get; set; } // Цена
}

public class Order {
    public int Id { get; set; }
    public DateTime OrderDate { get; set; }
    public int DrinkId { get; set; } // Внешний ключ
    public virtual Drink Drink { get; set; } // Навигационное свойство
}
```

#### 2. CRUD-операция (Создание заказа):

```
var newOrder = new Order { DrinkId = 1, OrderDate = DateTime.Now };  
db.Orders.Add(newOrder);  
db.SaveChanges();
```

**3. LINQ-запрос:** *Задача: Найти все заказы за сегодня, стоимостью выше 200 рублей, отсортированные по времени.*

```
var expensiveOrders = db.Orders  
    .Where(o => o.OrderDate.Date == DateTime.Today && o.Drink.Price > 200)  
    .OrderBy(o => o.OrderDate)  
    .Select(o => new { o.Drink.Name, o.OrderDate })  
    .ToList();
```

**4. Вывод:** Пользователю выводится список названий напитков и время их заказа для текущей смены.

## Лабораторная работа №1. Структурное моделирование информационной системы средствами IDEF0, IDEF3 и DFD

**Цель работы:** Сформировать у студентов навыки системного анализа предметной области, определения границ информационной системы и построения функциональных и потоковых моделей в нотациях *IDEF0*, *IDEF3* и *DFD*.

### Теоретические сведения

Структурный анализ базируется на методологиях:

- **IDEF0:** используется для создания функциональной модели, отображающей структуру и функции системы, а также потоки информации и материальных объектов. Главный принцип — декомпозиция функций.
- **IDEF3:** предназначена для описания логики выполнения процессов (сценариев), последовательности действий и условий их завершения/начала с использованием перекрестков.
- **DFD (Data Flow Diagrams):** описывает потоки данных между процессами и внешними сущностями, а также места хранения информации (хранилища данных).

### Порядок выполнения работы

1. Выбрать предметную область согласно индивидуальному варианту.
2. Описать назначение и цели проектируемой информационной системы.
3. Построить контекстную диаграмму **A-0** в нотации *IDEF0*.
4. Выполнить декомпозицию первого уровня (диаграмма **A0**), выделив 3–5 функциональных блоков.
5. Описать один из процессов диаграммы **A0** в логике **IDEF3**.
6. Составить текстовую или графическую **DFD-модель**, выделив внешние сущности и хранилища данных.
7. Оформить отчет согласно требованиям.

### Индивидуальные задания

№	Предметная область	Процесс для моделирования	Обязательные элементы
1	Библиотека	Выдача и возврат литературы	Книжный фонд, Читатели
2	Кофейня	Прием и приготовление заказа	Рецептура, Ингредиенты
3	Интернет-магазин	Обработка онлайн-заказов	Корзина, Платежный шлюз
4	Деканат	Ведение учета успеваемости	Студенты, Учебные планы
5	Поликлиника	Запись пациента на прием	Расписание, Медкарта

№	Предметная область	Процесс для моделирования	Обязательные элементы
6	Складской учёт	Приемка товаров от поставщика	Накладные, База остатков
7	Автосервис	Оказание услуг по ремонту	Мастера, Запчасти
8	Гостиница	Бронирование и регистрация	Номерной фонд, Гости
9	Фитнес-клуб	Оформление абонементов	Расписание, Тренеры
10	Доставка еды	Курьерская логистика	Курьеры, Карта города
11	Аптека	Продажа медикаментов	Рецепты, Фарм-контроль
12	Учебный центр	Формирование групп обучения	Договоры, Преподаватели
13	Турагентство	Бронирование турпакетов	Отели, Страховки
14	Ресторан	Резервирование столов	Схема зала, Меню
15	Сервисный центр	Диагностика и ремонт техники	Гарантия, Комплектующие
16	Авиакасса	Продажа авиабилетов	Рейсы, Паспортные данные
17	Отдел кадров	Найм новых сотрудников	Трудовой кодекс, Вакансии
18	Школьный журнал	Аттестация учащихся	Классы, Предметы
19	Прокат машин	Оформление аренды авто	Страховка, Водительские права
20	Техподдержка	Обработка обращений ( <i>Helpdesk</i> )	Тикеты, Инженеры

## Методические рекомендации

- Выбор процесса A-0:** Это должна быть самая общая функция системы (например, «Управлять продажами»).
- Входы vs Управления:** Вход (*I*) — это то, что «перерабатывается» процессом (данные клиента). Управление (*C*) — это правила (закон, инструкция), которые «направляют» процесс.
- Механизмы (M):** Ресурсы, которые выполняют работу, но не потребляются (персонал, сервер).
- Декомпозиция A0:** Разбейте систему на логические этапы (не более 6 блоков на одной диаграмме).
- Логика IDEF3:** Используйте перекрестки «И» (&), «ИЛИ» (O) и «Исключающее ИЛИ» (X) для ветвления процессов.
- DFD:** Следите за тем, чтобы данные не перемещались напрямую между внешними сущностями — они всегда должны проходить через процесс.

## Требования к отчёту

Отчет должен содержать:

1. Титульный лист.
2. Цель работы.
3. Описание предметной области (назначение ИС).
4. Контекстная диаграмма  $A - 0$  с описанием  $I, O, C, M$ .
5. Диаграмма декомпозиции  $A0$ .
6. Сценарий процесса в  $IDEF3$ .
7. Описание  $DFD$ -модели (сущности, процессы, хранилища).
8. Ответы на контрольные вопросы.

## Контрольные вопросы

1. В чем заключается основной принцип методологии  $IDEF0$ ?
2. Перечислите и опишите четыре типа стрелок в нотации  $IDEF0$ .
3. Что такое «декомпозиция» и для чего она применяется при проектировании ИС?
4. В чем заключается ключевое отличие нотации  $IDEF3$  от  $IDEF0$ ?
5. Какие типы перекрестков используются в диаграммах  $IDEF3$  и каково их назначение?
6. Назовите основные элементы  $DFD$ -диаграммы и опишите их функции.
7. Как в нотации  $DFD$  отображается место долговременного хранения информации?

## Образец решения. Вариант 0

**Предметная область:** деятельность кофейни.

### 1. Контекстная диаграмма $A-0$ :

- **Центральный блок:** «Обеспечить работу кофейни».
- **Входы ( $I$ ):** Клиенты, заказы, ингредиенты.
- **Управления ( $C$ ):** Меню, рецепты, нормы СанПиН.
- **Механизмы ( $M$ ):** Бариста, кассовый аппарат, кофемашина.
- **Выходы ( $O$ ):** Готовый напиток, кассовый чек, данные о выручке.

### 2. Декомпозиция $A0$ :

1. Принять оплату (Вход: Заказ → Выход: Чек).
2. Приготовить кофе (Управление: Рецепт → Механизм: Кофемашина).
3. Выдать заказ (Выход: Довольный клиент).

**3. IDEF3 (Процесс приготовления):** Помол зерен → Формирование кофейной таблетки → Экстракция → Взбивание молока → Смешивание ингредиентов.

**4. DFD-модель:**

- **Внешняя сущность:** Клиент.
- **Процесс:** Регистрация заказа.
- **Хранилище:** БД «Продажи».
- **Поток:** Сумма к оплате.

## Лабораторная работа №2. Проектирование базы данных и разработка простой информационной системы с использованием Entity Framework

**Цель работы:** Сформировать у студентов навыки проектирования реляционных баз данных, построения *ER*-моделей и реализации программного интерфейса для работы с данными с использованием технологии *EntityFramework* и языка запросов *LINQ*.

### Теоретические сведения

- **ER-модель (Entity-Relationship):** модель «сущность-связь», позволяющая описать концептуальную схему базы данных.
- **Entity Framework (EF):** современная технология *ORM* (Object-Relational Mapping) от Microsoft, которая позволяет разработчикам работать с реляционными данными с помощью объектов специфических для домена.
- **CRUD:** акроним для четырех базовых функций управления данными: *Create* (создание), *Read* (чтение), *Update* (обновление), *Delete* (удаление).
- **LINQ (Language Integrated Query):** язык запросов, встроенный в *C#*, позволяющий единообразно работать с различными источниками данных (коллекциями, базами данных).

### Порядок выполнения работы

1. Выбрать предметную область согласно варианту.
2. Разработать **ER-модель** (не менее 4 сущностей), определить первичные и внешние ключи.
3. Определить типы связей (1: 1, 1: *M*, *M*: *N*) и нормализовать схему.
4. Описать физическую структуру таблиц БД.
5. Создать программные модели классов на языке *C#* для *EntityFramework*.
6. Привести примеры реализации *CRUD*-операций для одной из сущностей.
7. Составить и описать сложный *LINQ*-запрос (с фильтрацией и сортировкой).
8. Оформить отчет и ответить на контрольные вопросы.

### Индивидуальные задания

№	Предметная область	Сущности (пример)	Задача для LINQ
1	Библиотека	Книга, Автор, Читатель, Выдача	Книги автора «X», изданные в 2020 году
2	Школьный журнал	Ученик, Учитель, Предмет, Оценка	Средний балл ученика по математике

№	Предметная область	Сущности (пример)	Задача для LINQ
3	Учет студентов	Студент, Группа, Кафедра, Дисциплина	Список студентов кафедры «ИТ» по алфавиту
4	Поликлиника	Врач, Пациент, Прием, Диагноз	Все приемы пациента «Иванов» за март
5	Интернет-магазин	Товар, Категория, Заказ, Клиент	Товары категории «Смартфоны» > 50 тыс. руб.
6	Складской учёт	Товар, Склад, Поставщик, Партия	Список пустых складов или остатки < 5 шт.
7	Гостиница	Номер, Тип_номера, Гость, Заселение	Свободные номера категории «Стандарт»
8	Фитнес-клуб	Клиент, Тренер, Абонемент, Посещение	Клиенты тренера «Петров» с активной картой
9	Автосервис	Авто, Клиент, Мастер, Заказ	Суммарная стоимость работ для авто «Toyota»
10	Доставка еды	Блюдо, Заказ, Курьер, Состав_заказа	Заказы, доставленные курьером №5 сегодня
11	Аптека	Препарат, Производитель, Чек, Кассир	Препараты производителя «Х» дешевле 100 руб.
12	Учебный центр	Курс, Преподаватель, Группа, Студент	Курсы, на которых обучается более 15 человек
13	Турфирма	Тур, Страна, Отель, Клиент	Самые дешевые туры в Египет с отелем 5*
14	Ресторан	Стол, Официант, Заказ, Блюдо	Выручка официанта «Сидоров» за смену
15	Сервисный центр	Устройство, Клиент, Заявка, Деталь	Все невыполненные заявки, принятые в понедельник
16	Продажа билетов	Рейс, Самолет, Билет, Пассажир	Список пассажиров на рейс «Москва-Сочи»
17	Кадровый учёт	Сотрудник, Отдел, Должность, Приказ	Сотрудники отдела кадров со стажем > 3 лет
18	Прокат авто	Машина, Клиент, Аренда, Тариф	Машины с АКПП, не находящиеся в аренде
19	Техподдержка	Заявка, Инженер, Статус, Категория	Срочные заявки, назначенные на инженера «У»
20	Учет проектов	Проект, Менеджер, Задача, Исполнитель	Задачи проекта «Альфа», срок которых истек

## Методические рекомендации

1. **Сущность vs Атрибут:** Сущность — это предмет или событие (например, «Книга»). Атрибут — свойство сущности («Название книги»).
2. **Связь M:N:** В *EF* связка «многие-ко-многим» реализуется через промежуточный класс, содержащий внешние ключи обеих таблиц.
3. **Первичный ключ (PK):** Всегда создавайте поле *Id* (тип *int*) для каждой модели.
4. **LINQ:** Используйте метод *.Include()* для подгрузки связанных данных и *.Where()* для условий.

## Требования к отчёту

1. Схема *ER*-модели (диаграмма сущностей).
2. Описание структуры таблиц (имена, типы данных, ключи).
3. Листинг программного кода классов-моделей (*C#*).
4. Примеры методов для *CRUD* операций.
5. Текст *LINQ*-запроса и описание его работы.

## Контрольные вопросы

1. Что такое первичный ключ (*Primary Key*) и какова его роль в базе данных?
2. Как реализуется связь «многие-ко-многим» в реляционных базах данных?
3. Объясните назначение технологии *Entity Framework*. Что означает подход *Code First*?
4. Какие компоненты входят в состав *CRUD*-операций и как они соотносятся с SQL-запросами?
5. Какую роль в *Entity Framework* играет класс, наследуемый от *DbContext*?
6. В чем преимущество использования *LINQ*-запросов перед обычными SQL-запросами?
7. Что такое «навигационные свойства» в моделях *Entity Framework* и для чего они нужны?

## Образец решения. Вариант 0

**Предметная область:** Библиотека.

**1. Сущности:** *Book* (Книга), *Author* (Автор), *Reader* (Читатель), *Loan* (Выдача). **2. Связи:** *Reader* — *Loan* (1: *M*), *Book* — *Loan* (1: *M*).

**3. Код модели (фрагмент):**

```
public class Book {
    public int Id { get; set; }
    public string Title { get; set; }
    public int Year { get; set; }
    // Навигационное свойство
    public virtual List<Loan> Loans { get; set; }
}
```

#### **4. CRUD (Create):**

```
context.Books.Add(new Book { Title = "C# Design Patterns", Year = 2022 });
context.SaveChanges();
```

#### **5. LINQ-запрос:**

```
// Найти все книги 2022 года, отсортированные по названию
var result = context.Books
    .Where(b => b.Year == 2022)
    .OrderBy(b => b.Title)
    .ToList();
```

Таблица 1 — Определение номера варианта контрольной работы

№ варианта	№ задания						
	Задание 1	Задание 2	Задание 3	Задание 4	Задание 5	Лабораторн ая работа №1	Лабораторн ая работа №2
00	10	9	10	8	9	10	9
01	4	16	6	8	17	4	16
02	10	1	18	2	15	10	1
03	17	17	18	15	4	17	17
04	17	16	12	10	20	17	16
05	16	6	11	18	5	16	6
06	13	4	19	2	5	13	4
07	1	18	16	17	15	1	18
08	5	17	19	1	8	5	17
09	8	9	18	7	12	8	9
10	17	10	4	7	18	17	10
11	14	8	1	12	11	14	8
12	10	13	9	19	10	10	13
13	6	9	3	3	20	6	9
14	16	3	17	18	20	16	3
15	15	7	3	8	8	15	7
16	5	6	15	20	13	5	6
17	2	15	9	9	16	2	15
18	7	2	3	3	3	7	2
19	18	9	9	11	13	18	9
20	13	11	14	14	14	13	11
21	7	2	13	20	8	7	2
22	6	18	3	14	12	6	18
23	7	14	1	11	12	7	14
24	5	3	6	19	9	5	3
25	10	8	3	9	16	10	8
26	11	9	8	8	9	11	9
27	3	2	3	8	10	3	2
28	15	20	12	8	19	15	20

Продолжение Таблицы 1.

№ варианта	№ задания						
	Задание 1	Задание 2	Задание 3	Задание 4	Задание 5	Лабораторн ая работа №1	Лабораторн ая работа №2
<b>29</b>	17	15	12	16	18	17	15
<b>30</b>	14	2	8	18	14	14	2
<b>31</b>	2	8	1	18	20	2	8
<b>32</b>	10	1	14	18	9	10	1
<b>33</b>	3	4	4	10	15	3	4
<b>34</b>	11	1	7	13	5	11	1
<b>35</b>	1	15	4	2	19	1	15
<b>36</b>	11	8	11	6	9	11	8
<b>37</b>	10	3	12	14	19	10	3
<b>38</b>	12	6	19	18	6	12	6
<b>39</b>	2	1	3	10	6	2	1
<b>40</b>	13	16	8	15	4	13	16
<b>41</b>	7	19	3	11	15	7	19
<b>42</b>	14	8	14	10	1	14	8
<b>43</b>	2	15	4	15	15	2	15
<b>44</b>	13	13	13	12	11	13	13
<b>45</b>	10	6	15	16	16	10	6
<b>46</b>	7	1	18	20	7	7	1
<b>47</b>	3	8	2	14	18	3	8
<b>48</b>	20	4	10	11	8	20	4
<b>49</b>	13	2	19	3	7	13	2
<b>50</b>	20	5	6	15	5	20	5
<b>51</b>	18	11	18	13	16	18	11
<b>52</b>	8	19	18	19	14	8	19
<b>53</b>	16	17	1	4	10	16	17
<b>54</b>	9	4	3	16	15	9	4
<b>55</b>	17	11	15	20	7	17	11
<b>56</b>	5	15	10	1	16	5	15
<b>57</b>	4	2	12	19	6	4	2
<b>58</b>	8	3	13	6	19	8	3

Продолжение Таблицы 1.

№ варианта	№ задания						
	Задание 1	Задание 2	Задание 3	Задание 4	Задание 5	Лабораторн ая работа №1	Лабораторн ая работа №2
<b>59</b>	13	2	18	7	5	13	2
<b>60</b>	14	20	13	11	8	14	20
<b>61</b>	13	15	2	9	11	13	15
<b>62</b>	12	12	17	7	13	12	12
<b>63</b>	14	9	18	7	3	14	9
<b>64</b>	18	1	4	9	1	18	1
<b>65</b>	17	5	13	17	6	17	5
<b>66</b>	8	10	16	17	17	8	10
<b>67</b>	6	4	17	16	12	6	4
<b>68</b>	16	3	7	7	14	16	3
<b>69</b>	11	12	7	3	15	11	12
<b>70</b>	17	15	2	8	13	17	15
<b>71</b>	12	11	3	3	14	12	11
<b>72</b>	3	18	3	9	3	3	18
<b>73</b>	8	16	12	1	19	8	16
<b>74</b>	9	12	13	16	15	9	12
<b>75</b>	5	14	17	1	11	5	14
<b>76</b>	20	19	4	17	13	20	19
<b>77</b>	10	5	1	1	10	10	5
<b>78</b>	6	12	5	14	18	6	12
<b>79</b>	7	5	4	17	3	7	5
<b>80</b>	18	20	19	18	3	18	20
<b>81</b>	12	9	1	2	14	12	9
<b>82</b>	3	6	6	7	4	3	6
<b>83</b>	14	8	7	7	3	14	8
<b>84</b>	6	14	4	19	15	6	14
<b>85</b>	13	15	18	4	6	13	15
<b>86</b>	20	4	4	11	19	20	4
<b>87</b>	17	13	1	8	1	17	13
<b>88</b>	18	7	20	3	6	18	7

Продолжение Таблицы 1.

№ варианта	№ задания						
	Задание 1	Задание 2	Задание 3	Задание 4	Задание 5	Лабораторн ая работа №1	Лабораторн ая работа №2
<b>89</b>	15	6	18	12	16	15	6
<b>90</b>	4	12	13	16	19	4	12
<b>91</b>	17	19	4	17	15	17	19
<b>92</b>	13	11	3	20	20	13	11
<b>93</b>	2	17	9	18	10	2	17
<b>94</b>	13	2	12	4	20	13	2
<b>95</b>	2	19	16	9	17	2	19
<b>96</b>	15	6	2	2	9	15	6
<b>97</b>	11	5	5	11	15	11	5
<b>98</b>	19	6	17	5	7	19	6
<b>99</b>	7	6	18	8	4	7	6