**Федеральное государственное бюджетное образовательное учреждение**

**высшего образования**

**«РОССИЙСКАЯ АКАДЕМИЯ НАРОДНОГО ХОЗЯЙСТВА   
И ГОСУДАРСТВЕННОЙ СЛУЖБЫ**

**ПРИ ПРЕЗИДЕНТЕ РОССИЙСКОЙ ФЕДЕРАЦИИ»**

**НИЖЕГОРОДСКИЙ ИНСТИТУТ УПРАВЛЕНИЯ – филиал РАНХиГС**

Факультет управления

Кафедра Информатики и информационных технологий

Специальность: 09.03.03 Прикладная информатика

Корпоративные информационные системы управления

**Отчет по лабораторным работам**

|  |  |
| --- | --- |
| по дисциплине: | Программная инженерия |
|  |  |
|  |  |
|  |  |

|  |  |  |
| --- | --- | --- |
|  |  | **АВТОР** |
|  |  | Обучающийся(иеся) 2 курса группы Ик-721 |
|  |  | заочной формы обучения |
|  |  |  |
|  |  | Беднякова Т. В. |
|  |  |  |
|  |  |  |
|  |
|  |
|  |
|  |
|  |
|  |
|  |
|  |

Нижний Новгород, 2025г.

# Лабораторная работа 1

Тема: «Освоение работы в Git, GitHab, GitLab»

Цель работы: Освоение основ работы с Git, GitHub и GitLab, а также знакомство с российской альтернативой GitFlick

## 1. Назначение, возможности, особенности.

1) Git:

Назначение: Git — это распределенная система контроля версий, предназначенная для отслеживания изменений в коде и совместной работы над проектами.

Возможности:

* Отслеживание изменений в файлах.
* Создание и управление ветками.
* Слияние изменений из разных веток.
* Возможность возврата к предыдущим версиям кода.
* Оффлайн-работа.

Особенности:

* Распределенная архитектура.
* Высокая скорость выполнения операций.
* Поддержка коммитов с сообщениями для документирования изменений.

2) GitHub:

Назначение: GitHub — это веб-сервис для хостинга Git-репозиториев с расширенными возможностями для совместной работы.

Возможности:

* Хостинг публичных и приватных репозиториев.
* Ведение обсуждений (issues) и управление проектами.
* Интеграция с CI/CD инструментами.
* Социальные функции: возможность следить за другими пользователями и проектами.

Особенности:

* Поддержка Markdown для документации.
* GitHub Actions для автоматизации процессов.
* Удобный веб-интерфейс для управления репозиториями.

3) GitLab:

Назначение: GitLab — это веб-сервис для хостинга Git-репозиториев с акцентом на DevOps и управление проектами.

Возможности:

* Полный цикл разработки: от планирования до развертывания (CI/CD).
* Хостинг публичных и приватных репозиториев.
* Управление задачами и создание вики для документации.

Особенности:

* Глубокая интеграция инструментов DevOps.
* Возможность использования GitLab CI для автоматизации процессов.
* Разнообразные уровни доступа для пользователей.

## 2. Сравнение (Git, GitHub vs GitLab).

Функциональность:

* GitHub фокусируется на социальном взаимодействии и совместной работе.
* GitLab предлагает более полное решение DevOps с интегрированными CI/CD инструментами.

Хостинг:

* GitHub предоставляет облачный хостинг, но также есть возможность локального развертывания через GitHub Enterprise.
* GitLab предлагает как облачный, так и локальный хостинг, что делает его более гибким для корпоративного использования.

Интерфейс:

* Оба имеют удобные веб-интерфейсы, но предпочтения могут зависеть от личного опыта разработчика.

Цены:

* GitHub имеет бесплатный тарифный план с ограничениями на приватные репозитории, а платные планы могут быть дорогими для больших команд.
* GitLab предлагает более щедрые бесплатные тарифы и различные уровни платных планов.

Сообщество и экосистема:

* GitHub имеет огромное сообщество и множество интеграций с другими сервисами.
* GitLab также активно развивается и имеет свою преданную аудиторию.

Выбор между Git, GitHub и GitLab зависит от конкретных потребностей вашего проекта или команды. Если вам нужен просто инструмент контроля версий — используйте Git. Если вы хотите хостить свои репозитории и работать в команде — выбирайте между GitHub и GitLab в зависимости от ваших требований к функциональности и интеграции.

## 3. Российская система GitFlick.

GitFlick — это российская платформа для хостинга Git-репозиториев, которая предоставляет возможности для совместной работы над проектами. Она ориентирована на пользователей из России и СНГ и предлагает функционал, схожий с GitHub и GitLab, включая:

* Хостинг репозиториев.
* Инструменты для управления проектами и задачами.
* Интеграцию с CI/CD процессами.
* Поддержку локализации и соответствие требованиям законодательства РФ.

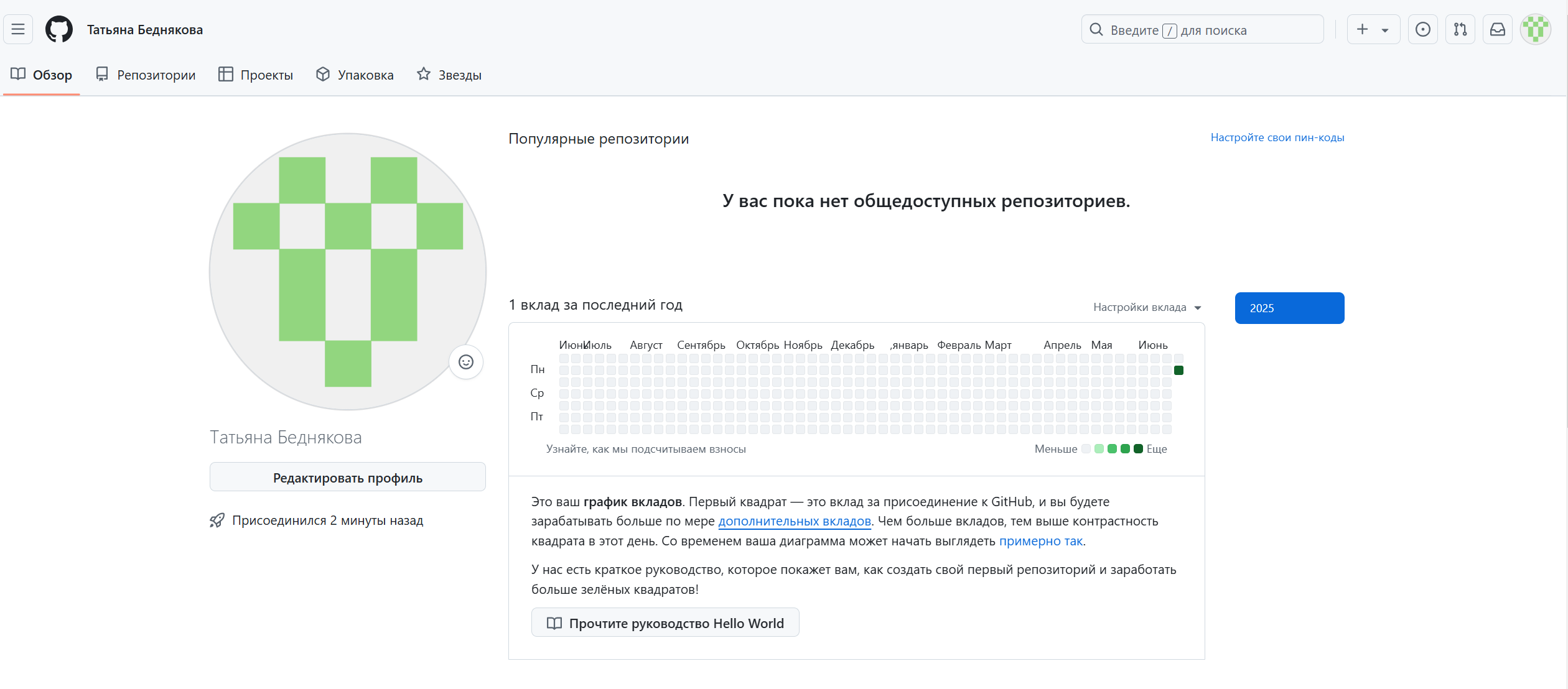
## 4. Команды системы Git: названия и назначение.

Основные команды Git и их назначение:

* git init — Инициализация нового репозитория.
* git clone <url> — Клонирование удаленного репозитория на локальную машину.
* git add <file> — Добавление изменений в индекс (staging area).
* git commit -m "<message>" — Фиксация изменений с сообщением.
* git status — Проверка состояния рабочего каталога и индекса.
* git log — Просмотр истории коммитов.
* git branch — Просмотр существующих веток или создание новой ветки (git branch <branch-name>).
* git checkout <branch-name> — Переключение на другую ветку.
* git merge <branch-name> — Слияние указанной ветки с текущей.
* git pull — Получение изменений из удаленного репозитория и их слияние с текущей веткой.
* git push — Отправка локальных коммитов в удаленный репозиторий.

Эти команды образуют основу работы с Git и позволяют эффективно управлять версиями кода в проектах.

## 5. Зарегистрироваться в одном из репозиториев и записать в них отчёты по лабораторным работам.

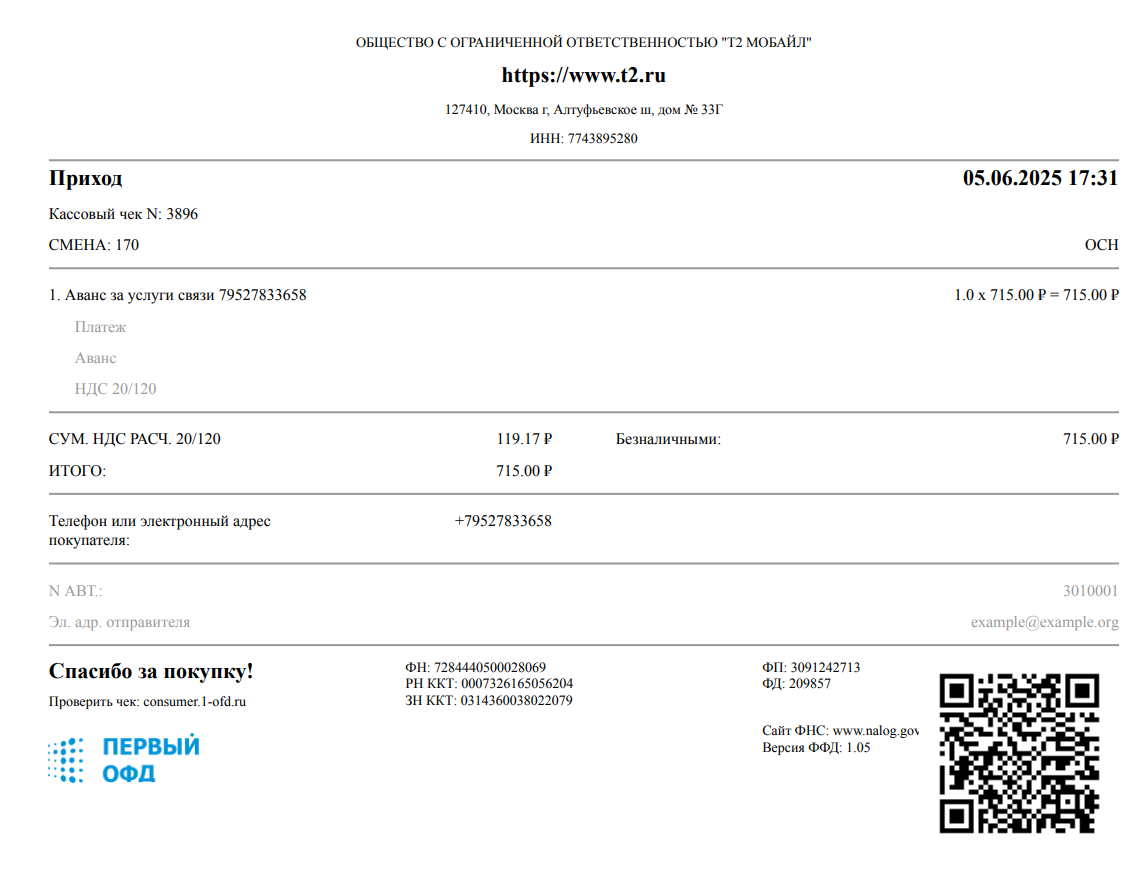


# Лабораторная работа 2

Тема: «Анализ экономического документа»

Цель работы: проведение комплексного анализа экономического документа через выполнение реквизитного анализа, выделение финансовых и экономических реквизитов с определением их оснований и признаков. Также описание процесса работы с данным документом, включая его функциональное применение в бизнес-процессах, и визуализация схемы работы с использованием нотации BPMN.

## 1. Выбрать любой документ.



## 2. Провести реквизитный анализ.

Основные реквизиты кассового чека:

1. Наименование продавца : ООО "Т2 МОБАЙЛ"
2. ИНН продавца : 7743895280
3. Адрес расчетов или место нахождения продавца : Москва г, Алтуфьевское ш, дом № 33Г
4. Дата и время совершения покупки : 05.06.2025 17:31
5. Порядковый номер чека : 3896
6. Система налогообложения : ОСН
7. Информация о товаре/услуге :

* Наименование: Аванс за услуги связи
* Количество: 1.0
* Цена: 715.00 ₽
* Стоимость: 715.00 ₽

1. Сумма НДС (ставка 20%) : 119.17 ₽
2. Итоговая сумма платежа : 715.00 ₽
3. Форма оплаты : Безналичными
4. Электронная подпись ФН : 3091242713
5. Фискальный номер устройства (ФН) : 7284440500028069
6. Номер фискального документа (ФД) : 209857
7. Завершающий реквизит (QR-код или ссылка для проверки чека) : consumer.1-ofd.ru
8. Контактная информация покупателя : +79527833658
9. Электронный адрес отправителя чека : [example@example.org](mailto:example@example.org)

## 3. Выделить финансовые и экономический реквизиты, определить в них основание и признак.

|  |  |  |
| --- | --- | --- |
| **Реквизит** | **Основание** | **Признак** |
| Сумма аванса | Общая стоимость операции | 715.00 ₽ |
| Сумма НДС (20%) | Налоговая нагрузка | 119.17 ₽ |
| Форма оплаты | Метод расчёта | Безналичный |
| Дата и время | Время совершения операции | 05.06.2025 17:31 |
| Телефон клиента | Идентификатор клиента | +79527833658 |
| Количество | Объём товара/услуги | 1.0 ед. |
| Цена за единицу | Стоимость одной позиции | 715.00 ₽ |

## 4. Создать соответствующую базу данных –сначала концептуальную, логическую, физическую, позволяющую работать с этим документом.

1) Концептуальная модель

Отражает основные сущности и связи между ними без детализации полей.

Сущности:

* Продавец – организация, принимающая оплату.
* Чек – документ, фиксирующий факт оплаты.
* Покупатель – клиент, совершающий оплату.
* Фискальные данные – данные, связанные с ККТ и ФН.
* Операция – детали оплаты (наименование, цена, кол-во).
* Налоги – информация о налоговых начислениях.

Связи:

* Продавец → Чек
* Покупатель ↔ Чек
* Чек ↔ Операция
* Чек ↔ Фискальные данные
* Операция ↔ Налоги

2) Логическая модель

Уточняет атрибуты сущностей и связи.

Таблицы:

* sellers – id, name, inn, address
* customers – id, phone, email
* receipts – id, seller\_id, customer\_id, datetime, shift\_number, tax\_system, total\_sum, payment\_type
* operations – id, receipt\_id, item\_name, quantity, price\_per\_unit, total\_price
* taxes – id, operation\_id, tax\_rate, tax\_amount
* fiscal\_data – id, receipt\_id, fn\_number, fd\_number, fp\_signature, fiscal\_sign, ofd\_link

3) Физическая модель

SQL-скрипт создания таблиц:

CREATE TABLE sellers (

seller\_id INT PRIMARY KEY,

name VARCHAR(255),

inn VARCHAR(12),

address VARCHAR(255)

);

CREATE TABLE customers (

customer\_id INT PRIMARY KEY,

phone VARCHAR(20),

email VARCHAR(100)

);

CREATE TABLE receipts (

receipt\_id INT PRIMARY KEY,

seller\_id INT,

customer\_id INT,

datetime DATETIME,

shift\_number INT,

tax\_system VARCHAR(50),

total\_sum DECIMAL(10,2),

payment\_type VARCHAR(50),

FOREIGN KEY (seller\_id) REFERENCES sellers(seller\_id),

FOREIGN KEY (customer\_id) REFERENCES customers(customer\_id)

);

CREATE TABLE operations (

operation\_id INT PRIMARY KEY,

receipt\_id INT,

item\_name VARCHAR(255),

quantity DECIMAL(10,2),

price\_per\_unit DECIMAL(10,2),

total\_price DECIMAL(10,2),

FOREIGN KEY (receipt\_id) REFERENCES receipts(receipt\_id)

);

CREATE TABLE taxes (

tax\_id INT PRIMARY KEY,

operation\_id INT,

tax\_rate VARCHAR(10),

tax\_amount DECIMAL(10,2),

FOREIGN KEY (operation\_id) REFERENCES operations(operation\_id)

);

CREATE TABLE fiscal\_data (

fiscal\_id INT PRIMARY KEY,

receipt\_id INT,

fn\_number VARCHAR(50),

fd\_number VARCHAR(50),

fp\_signature VARCHAR(50),

fiscal\_sign VARCHAR(50),

ofd\_link VARCHAR(255),

FOREIGN KEY (receipt\_id) REFERENCES receipts(receipt\_id)

);

## 5. Описать процесс работы с этим документом. Отобразить схему работы с ним с помощью нотации BPMN.

Описание этапов обработки кассового чека:

1. Создание чека – формируется в момент оплаты.
2. Передача чека в ОФД – через интернет передается в оператор фискальных данных.
3. Фискализация чека – чек регистрируется в системе ФНС через ОФД.
4. Хранение чека – сохраняется в электронном виде у организации и ОФД.
5. Обработка в учетной системе – импорт данных в ERP-систему (например, 1С).
6. Анализ данных – финансовый и налоговый учет.
7. Проверка покупателем – клиент может проверить чек по ссылке или QR-коду.

BPMN диаграмма (описание в текстовом виде)

[Начало] → [Создание чека]

↓

[Передача в ОФД]

↓

[Фискализация чека]

↓

[Хранение чека]

↓

[Импорт в ERP]

↓

[Финансовый учет]

↓

[Проверка покупателем]

↓

[Конец процесса]

# Лабораторная работа 3

Тема: «Исследование общероссийских классификаторов (по книге Грекула)»

Цель работы: Изучить структуру и назначение общероссийских классификаторов, освоить формулу кодирования, определить характеристики на примере двух классификаторов и спроектировать базу данных для их хранения и использования.

## 1. Типы классификаторов, назначение.

Общероссийские классификаторы — это нормативные документы, содержащие систематизированные данные для кодирования и обработки информации в различных сферах.

Основные типы классификаторов:

* Классификаторы продукции (ОКП): Используются для кодирования товаров и услуг в целях статистики, таможенного оформления и учета.
* Классификаторы организаций и предприятий (ОКПО): Присвоение уникальных кодов всем организациям и индивидуальным предпринимателям.
* Классификаторы профессий (ОКПДТР): Кодирование видов трудовой деятельности.
* Классификаторы отраслей экономики (ОКОНХ): Группировка хозяйствующих субъектов по отраслям.
* Классификаторы территориальных единиц (ОКТМО, ОКАТО): Учет административно-территориального деления РФ.
* Классификаторы валют и денежных единиц (ОКВ): Кодирование валют для международной статистики и расчетов.

## 2. О кодировании классификаторов (формула классификатора).

1) Иерархическая система кодирования

Предполагает последовательное деление множества объектов на подчиненные группы.

Формула:

K = A × B × C , где:

* A – число разрядов,
* B – основание системы кодирования,
* C – количество возможных комбинаций.

Пример: ОКПД2 — Общероссийский классификатор продукции по видам экономической деятельности.

Структура: 5 разрядов → раздел, класс, подкласс, группу, подгруппу.

2) Фасетная система кодирования

Разделение признаков объектов на независимые категории (фасеты).

Применяется редко из-за сложности реализации.

## 3. Характеристики классификаторов для двух выбранных классификаторов.

1) ОКПО (Общероссийский классификатор предприятий и организаций)

* Назначение : присвоение уникального идентификационного номера каждому юридическому лицу и ИП.
* Длина кода : 8 знаков
* Уровень детализации : уникальный код организации
* Использование : в государственной статистике, налоговых органах, в системе госзакупок
* Пример : 7743895280 — ИНН ООО "Т2 МОБАЙЛ".

2) ОКТМО (Общероссийский классификатор территорий муниципальных образований)

* Назначение : кодирование муниципальных образований для сбора статистических данных
* Длина кода : 8–11 знаков
* Уровень детализации : регион, район, городское/сельское поселение
* Использование : бюджетная сфера, местное самоуправление, перепись населения
* Пример : если адрес кассового чека – Москва г, Алтуфьевское ш, дом № 33Г, то ему соответствует определенный ОКТМО-код, например: 45901004 (условно)

## 4. Схема БД для выбранных классификаторов.

Концептуальная модель

Сущности:

* Organization – организация
* OKPO – запись из ОКПО
* Address – адрес организации
* OKTMO – запись из ОКТМО

Связи:

* Organization ↔ OKPO (один к одному)
* Organization ↔ Address (один к одному)
* Address ↔ OKTMO (один к одному)

Физическая модель (SQL DDL)

CREATE TABLE OKPO (

okpo\_code CHAR(8) PRIMARY KEY,

full\_name VARCHAR(255),

short\_name VARCHAR(100),

inn CHAR(12)

);

CREATE TABLE OKTMO (

oktmo\_code CHAR(11) PRIMARY KEY,

region VARCHAR(100),

district VARCHAR(100),

settlement VARCHAR(100),

name VARCHAR(255)

);

CREATE TABLE Address (

address\_id INT PRIMARY KEY,

street VARCHAR(100),

building VARCHAR(10),

city VARCHAR(50),

oktmo\_code CHAR(11),

FOREIGN KEY (oktmo\_code) REFERENCES OKTMO(oktmo\_code)

);

CREATE TABLE Organization (

organization\_id INT PRIMARY KEY,

okpo\_code CHAR(8),

address\_id INT,

FOREIGN KEY (okpo\_code) REFERENCES OKPO(okpo\_code),

FOREIGN KEY (address\_id) REFERENCES Address(address\_id)

);