

Документ подписан простой электронной подписью
Информация о владельце:
ФИО: Комарова Светлана Юриевна
Должность: Проректор по образовательной деятельности
Дата подписания: 30.07.2026 21:16:59
Уникальный программный ключ:
43ba42f5deae4116bbfcb9ac98e391080312f5e541104d4c47b9e031

Приложение

**Федеральное государственное бюджетное образовательное
учреждение высшего образования
«Омский государственный аграрный университет
имени П.А. Столыпина»**

Университетский колледж агробизнеса

09.02.11 Разработка и управление программным обеспечением

**ФОНД ОЦЕНОЧНЫХ СРЕДСТВ
по ПП.01.01 Производственная практика (по профилю специальности)**

Обеспечивающее преподавание дисциплины
подразделение

Инженерное отделение

Разработчик:

Преподаватель

А.В. Кортусов

**Омск
2026**

СОДЕРЖАНИЕ

1. ОБЩИЕ ПОЛОЖЕНИЯ	3
2. ОЖИДАЕМЫЕ РЕЗУЛЬТАТЫ ИЗУЧЕНИЯ	4
3. РАСПРЕДЕЛЕНИЕ ОЦЕНИВАНИЯ РЕЗУЛЬТАТОВ ОБУЧЕНИЯ И ТИПОВ ОЦЕНОЧНЫХ МАТЕРИАЛОВ ПО ЭЛЕМЕНТАМ ЗНАНИЙ И УМЕНИЙ	6
4. ПОКАЗАТЕЛИ ОЦЕНКИ РЕЗУЛЬТАТОВ ОСВОЕНИЯ ПРОИЗВОДСТВЕННОЙ ПРАКТИКИ	7

1. ОБЩИЕ ПОЛОЖЕНИЯ

1. Фонд оценочных средств (далее – ФОС) предназначен для контроля и оценки образовательных достижений обучающихся, освоивших программу ПП.01.01 Производственная практика (по профилю специальности).
2. ФОС включает оценочные материалы для проведения текущего контроля и промежуточной аттестации в форме зачета.
3. ФОС позволяет оценивать знания, умения, направленные на формирование компетенций.
4. ФОС разработан на основании положений основной образовательной программы по специальности 09.02.11 Разработка и управление программным обеспечением и рабочей программы ПП.01.01 Производственная практика (по профилю специальности).
5. ФОС является обязательным обособленным приложением к рабочей программе.

2. ОЖИДАЕМЫЕ РЕЗУЛЬТАТЫ ИЗУЧЕНИЯ

Результаты обучения (освоенные навыки, умения)	Показатели оценки образовательных результатов
разработки концептуальной модели базы данных, инфологической модели базы данных, разработки физической модели базы данных	Обучающийся обладает навыками разработки концептуальной модели базы данных, инфологической модели базы данных, разработки физической модели базы данных
разработки требований к базе данных, нормализация структуры базы данных	Обучающийся обладает навыками разработки требований к базе данных, нормализация структуры базы данных
работы с различными объектами базы данных	Обучающийся обладает навыками работы с различными объектами базы данных
создания таблиц базы данных с определением структуры и типов данных для каждого атрибута	Обучающийся обладает навыками создания таблиц базы данных с определением структуры и типов данных для каждого атрибута
оптимизации запросов для повышения производительности системы	Обучающийся обладает навыками оптимизации запросов для повышения производительности системы
создания и удаления баз данных, восстановления баз данных, резервного копирования баз данных	Обучающийся обладает навыками создания и удаления баз данных, восстановления баз данных, резервного копирования баз данных
создания пользователей и назначения прав доступа	Обучающийся обладает навыками создания пользователей и назначения прав доступа
использования стандартных методов защиты объектов базы данных	Обучающийся обладает навыками использования стандартных методов защиты объектов базы данных
разработки и внедрения систем защиты баз данных от несанкционированного доступа разработки и внедрения систем резервного копирования и восстановления баз данных аудита безопасности баз данных	Обучающийся обладает навыками разработки и внедрения систем защиты баз данных от несанкционированного доступа разработки и внедрения систем резервного копирования и восстановления баз данных аудита безопасности баз данных
определять требования к базе данных	Обучающийся умеет определять требования к базе данных
разрабатывать концептуальную, логическую и физическую модели баз данных	Обучающийся умеет разрабатывать концептуальную, логическую и физическую модели баз данных
проектировать схему базы данных	Обучающийся умеет проектировать схему базы данных
разрабатывать объекты баз данных создавать таблицы, индексы, ограничения и другие объекты базы данных	Обучающийся умеет разрабатывать объекты баз данных создавать таблицы, индексы, ограничения и другие объекты базы данных
разрабатывать хранимые процедуры и триггеры для баз данных	Обучающийся умеет разрабатывать хранимые процедуры и триггеры для баз данных
разрабатывать необходимые для различных групп пользователей представления	Обучающийся умеет разрабатывать необходимые для различных групп пользователей представления

программировать и создавать хранимые процедуры, функции и триггеры для обработки данных	Обучающийся умеет программировать и создавать хранимые процедуры, функции и триггеры для обработки данных
оптимизировать запросы и проводить мониторинг производительности базы данных	Обучающийся умеет оптимизировать запросы и проводить мониторинг производительности базы данных
создавать и удалять базы данных	Обучающийся умеет создавать и удалять базы данных
создавать пользователей и назначать права доступа;	Обучающийся умеет создавать пользователей и назначать права доступа;
управлять транзакциями и контролировать целостность данных;	Обучающийся умеет управлять транзакциями и контролировать целостность данных;
обеспечивать безопасность и управлять доступом к данным;	Обучающийся умеет обеспечивать безопасность и управлять доступом к данным;
разрабатывать и внедрять системы защиты баз данных от несанкционированного доступа	Обучающийся умеет разрабатывать и внедрять системы защиты баз данных от несанкционированного доступа
разрабатывать и внедрять системы резервного копирования и восстановления баз данных	Обучающийся умеет разрабатывать и внедрять системы резервного копирования и восстановления баз данных
устанавливать и настраивать механизмы аутентификации и авторизации пользователей, создавать и управлять ролями и правами доступа к данным	Обучающийся умеет устанавливать и настраивать механизмы аутентификации и авторизации пользователей, создавать и управлять ролями и правами доступа к данным

3. РАСПРЕДЕЛЕНИЕ ОЦЕНИВАНИЯ РЕЗУЛЬТАТОВ ОБУЧЕНИЯ И ТИПОВ ОЦЕНОЧНЫХ МАТЕРИАЛОВ ПО ЭЛЕМЕНТАМ УМЕНИЙ И НАВЫКОВ

Содержание курса	Форма контроля	Умения	Навыки
Текущий контроль			
Раздел 1. Организационный этап			
Прохождение вводного инструктажа. Получение и обсуждение задания на практику.	Проверка отчета по учебной практике	Уо 02.01, Уо 06.01	-
Раздел 2. Основной этап			
Решение прикладных задач:	наблюдение и оценка в процессе практики; анализ отчетной документации; экспертная оценка выполнения индивидуальных заданий.	У 1.1.1, У 1.1.2, У 1.1.3, У 1.2.1, У 1.2.2, У 1.2.3, У 1.3.1, У 1.3.2, У 1.4.1, У 1.4.2, У 1.4.3, У 1.4.4У 1.5.1, У 1.5.2, У 1.5.3	Н 1.1.1, Н 1.1.2, Н 1.2.1, Н 1.3.1, Н 1.3.2, Н 1.4.1, Н 1.4.2, Н 1.5.1, Н 1.5.2
Раздел 3. Заключительный этап			
Оформление введения. Оформление основной части. Оформление заключения. Оформление списка использованных источников и приложений. Оформление отчета и приложений. Прохождение собеседования (зачет)	Проверка отчета по учебной практике, собеседование	Уо 02.01, Уо 06.01	-
Промежуточный контроль			
Зачет	Проверка отчета по учебной практике, собеседование	У 1.1.1, У 1.1.2, У 1.1.3, У 1.2.1, У 1.2.2, У 1.2.3, У 1.3.1, У 1.3.2, У 1.4.1, У 1.4.2, У 1.4.3, У 1.4.4У 1.5.1, У 1.5.2, У 1.5.3, Уо 02.01, Уо 06.01	Н 1.1.1, Н 1.1.2, Н 1.2.1, Н 1.3.1, Н 1.3.2, Н 1.4.1, Н 1.4.2, Н 1.5.1, Н 1.5.2

4. МЕТОДИЧЕСКИЕ МАТЕРИАЛЫ ДЛЯ ОЦЕНКИ УМЕНИЙ, НАВЫКОВ

4.1. Оценочные средства, применяемые для текущего контроля.

4.1.1. Разработка структуры базы данных организации

На основе описания предметной области создайте структуру базы данных организации в СУБД:

1. Проведите анализ предметной области и выделите основные сущности.
2. Определите атрибуты каждой сущности.
3. Разработайте ER-диаграмму предметной области.
4. Выполните нормализацию базы данных до третьей нормальной формы (3НФ).
5. Создайте SQL-скрипты для создания таблиц, ограничений, индексов.
6. Заполните таблицы тестовыми данными (не менее 15-20 записей в каждой таблице).

Вариант задания: База данных стоматологической клиники

На основе анализа предметной области выделены следующие сущности:

Сущность	Назначение	Основные атрибуты
Пациенты	Хранение информации о пациентах	Номер карточки (РК), ФИО, дата рождения, пол, телефон, адрес, медицинский полис (с маской ввода АБ-1234)
Врачи	Хранение информации о врачах	Табельный номер (РК), ФИО, должность, специализация, телефон, кабинет
Услуги	Перечень медицинских услуг	Код услуги (РК), наименование, стоимость, средняя продолжительность
Приемы	Журнал приемов пациентов	Номер приема (РК), номер карточки (FK), табельный номер врача (FK), код услуги (FK), дата, время, стоимость, статус

Должности	Справочник должностей	Код должности (PK), наименование, обязанности оклад,
-----------	-----------------------	---

Пример SQL-скрипта с маской ввода:

-- Создание таблицы пациентов

```
CREATE TABLE Пациенты (
    НомерКарточки INT PRIMARY KEY AUTO_INCREMENT,
    ФИО VARCHAR(100) NOT NULL,
    ДатаРождения DATE NOT NULL,
    Пол ENUM('Мужской', 'Женский') DEFAULT 'Мужской',
    Телефон VARCHAR(20),
    Адрес VARCHAR(200),
    МедПолис VARCHAR(10) UNIQUE NOT NULL
);
```

-- Создание таблицы врачей

```
CREATE TABLE Врачи (
    ТабельныйНомер INT PRIMARY KEY AUTO_INCREMENT,
    ФИО VARCHAR(100) NOT NULL,
    КодДолжности INT NOT NULL,
    Специализация VARCHAR(100),
    Телефон VARCHAR(20),
    Кабинет VARCHAR(10),
    FOREIGN KEY (КодДолжности) REFERENCES Должности(КодДолжности)
);
```

Форма выполнения: Отчет с ER-диаграммой, SQL-скриптами создания БД и заполнения данными, обоснованием нормализации.

4.1.2. Разработка объектов базы данных

Создайте следующие объекты базы данных в соответствии с разработанной структурой:

1. Хранимые процедуры:

- Создайте хранимую процедуру для записи нового приема с проверкой занятости врача.
- Создайте процедуру для формирования ежедневного отчета по приемам.
- Реализуйте процедуру для изменения статуса приема (запланирован, выполнен, отменен).

2. Триггеры:

- Создайте триггер для автоматического обновления общей стоимости приема при изменении услуги.
- Реализуйте триггер для логирования всех изменений в таблице пациентов.
- Создайте триггер для проверки корректности медицинского полиса при вставке/обновлении.

3. Функции:

- Создайте функцию для расчета общей стоимости приемов пациента за период.
- Реализуйте функцию для проверки доступности врача на заданное время.

Пример реализации процедуры и триггера:

-- Хранимая процедура записи на прием

CREATE PROCEDURE ЗаписатьНаПрием (

IN р_номер_карточки INT,

IN р_табельный_номер INT,

IN р_код_услуги INT,

IN р_дата DATE,

IN р_время TIME

)

BEGIN

DECLARE занято INT;

-- Проверка занятости врача

SELECT COUNT(*) INTO занято

FROM Приемы

WHERE ТабельныйНомер = р_табельный_номер

AND Дата = р_дата

AND Время = р_время

AND Статус != 'Отменен';

IF занято = 0 THEN

INSERT INTO Приемы (НомерКарточки, ТабельныйНомер, КодУслуги, Дата, Время, Статус)

VALUES (р_номер_карточки, р_табельный_номер, р_код_услуги, р_дата, р_время, 'Запланирован');

ELSE

SIGNAL SQLSTATE '45000' SET MESSAGE_TEXT = 'Врач занят в указанное время';

END IF;

END;

-- Триггер для логирования изменений пациентов

CREATE TRIGGER ЛогированиеПациентов

AFTER UPDATE ON Пациенты

FOR EACH ROW

BEGIN

INSERT INTO ЖурналИзменений (Таблица, Действие, СтараяЗапись, НоваяЗапись, Дата)

VALUES ('Пациенты', 'UPDATE',

CONCAT('ФИО:', OLD.ФИО, '; Телефон:', OLD.Телефон),

CONCAT('ФИО:', NEW.ФИО, '; Телефон:', NEW.Телефон),

NOW());

END;

Форма выполнения: SQL-скрипты с процедурами, функциями и триггерами, а также примеры их вызова и демонстрация работы.

4.1.3. Реализация безопасности баз данных

Разработайте систему защиты базы данных в соответствии с требованиями безопасности:

1. Управление пользователями:

- Создайте роли пользователей: Администратор, Регистратор, Врач.
 - Создайте учетные записи пользователей с назначением соответствующих ролей.
 - Настройте права доступа для каждой роли.
2. Шифрование данных:
- Реализуйте шифрование личных данных пациентов (ФИО, телефон, адрес).
 - Создайте функцию шифрования и дешифрования для полей с конфиденциальной информацией.
3. Резервное копирование:
- Создайте скрипт автоматического резервного копирования базы данных.
 - Настройте регулярное выполнение резервного копирования.

Пример настройки прав доступа и шифрования:

sql

-- Создание ролей

```
CREATE ROLE Администратор;
CREATE ROLE Регистратор;
CREATE ROLE Врач;
```

-- Назначение прав

```
GRANT ALL PRIVILEGES ON стоматология.* TO Администратор;
GRANT SELECT, INSERT, UPDATE ON стоматология.Пациенты TO Регистратор;
GRANT SELECT, UPDATE ON стоматология.Пациенты TO Врач;
GRANT SELECT, INSERT, UPDATE ON стоматология.Приемы TO Регистратор;
GRANT SELECT ON стоматология.Приемы TO Врач;
```

-- Создание функций шифрования (пример для MySQL)

```
CREATE FUNCTION Шифровать(текст VARCHAR(255), ключ VARCHAR(50))
RETURNS VARBINARY(255)
DETERMINISTIC
BEGIN
    RETURN AES_ENCRYPT(текст, ключ);
END;
```

```
CREATE FUNCTION Дешифровать(данные VARBINARY(255), ключ VARCHAR(50))
RETURNS VARCHAR(255)
DETERMINISTIC
BEGIN
    RETURN AES_DECRYPT(данные, ключ);
END;
```

Форма выполнения: Скрипты создания пользователей и ролей, настройки прав доступа, функции шифрования, регламент резервного копирования.

4.1.4. Мониторинг и оптимизация производительности

Выполните анализ и оптимизацию работы базы данных:

1. Анализ запросов:
 - Выполните анализ 5-10 часто выполняемых запросов с использованием EXPLAIN.
 - Определите проблемные запросы и узкие места.
2. Оптимизация:
 - Создайте необходимые индексы для ускорения выполнения запросов.

- Оптимизируйте структуру таблиц (разбиение, архивирование старых данных).
 - Оптимизируйте написание сложных запросов.
3. Мониторинг:
- Настройте сбор статистики выполнения запросов.
 - Разработайте скрипт для отслеживания ключевых показателей производительности.

Пример анализа и оптимизации запросов:

-- Анализ запроса на получение количества приемов по врачам за период

```
EXPLAIN SELECT Врачи.ФИО, COUNT(*) as КоличествоПриемов
FROM Приемы
JOIN Врачи ON Приемы.ТабельныйНомер = Врачи.ТабельныйНомер
WHERE Приемы.Дата BETWEEN '2026-01-01' AND '2026-01-31'
GROUP BY Врачи.ТабельныйНомер;
```

-- Создание индексов для ускорения запроса

```
CREATE INDEX idx_приемы_дата ON Приемы(Дата);
CREATE INDEX idx_приемы_врач ON Приемы(ТабельныйНомер);
CREATE INDEX idx_приемы_статус ON Приемы(Статус);
```

-- Создание представления для частого использования

```
CREATE VIEW ОтчетПоВрачам AS
SELECT Врачи.ФИО,
       Врачи.Специализация,
       COUNT(Приемы.НомерПриема) as ВсегоПриемов,
       SUM(Услуги.Стоимость) as ОбщаяВыручка
FROM Врачи
LEFT JOIN Приемы ON Врачи.ТабельныйНомер = Приемы.ТабельныйНомер
LEFT JOIN Услуги ON Приемы.КодУслуги = Услуги.КодУслуги
WHERE Приемы.Статус = 'Выполнен'
GROUP BY Врачи.ТабельныйНомер;
```

Форма выполнения: Отчет с анализом запросов, созданными индексами, оптимизированными запросами и рекомендациями по улучшению производительности.

4.1.5. Администрирование и защита базы данных

Разработайте комплекс мер по администрированию и защите базы данных:

1. Политика резервного копирования:
 - Разработайте регламент резервного копирования (полное, дифференциальное, инкрементное).
 - Создайте скрипты для автоматического резервного копирования.
 - Настройте хранение и ротацию архивов.
2. Мониторинг безопасности:
 - Создайте журнал аудита для отслеживания подозрительных действий.
 - Реализуйте механизм оповещения о попытках несанкционированного доступа.
 - Настройте мониторинг активности пользователей.
3. Восстановление базы данных:
 - Разработайте план восстановления базы данных в аварийных ситуациях.
 - Создайте скрипты для восстановления данных из резервных копий.

Пример скрипта для резервного копирования и аудита:

-- Создание таблицы для аудита

```
CREATE TABLE ЖурналАудита (  
  id INT PRIMARY KEY AUTO_INCREMENT,  
  Пользователь VARCHAR(50),  
  Действие VARCHAR(100),  
  Таблица VARCHAR(50),  
  Данные TEXT,  
  IP_Адрес VARCHAR(45),  
  Время TIMESTAMP DEFAULT CURRENT_TIMESTAMP  
);
```

-- Триггер для аудита изменений

```
CREATE TRIGGER ТриггерАудитаПриемы  
AFTER UPDATE ON Приемы  
FOR EACH ROW  
BEGIN  
  INSERT INTO ЖурналАудита (Пользователь, Действие, Таблица, Данные)  
  VALUES (CURRENT_USER(), 'UPDATE', 'Приемы',  
    CONCAT('Было: ', OLD.НомерПриема, '->', OLD.Статус, ' Стало: ',  
    NEW.Статус));  
END;
```

-- Скрипт резервного копирования (пример для MySQL)

```
CREATE PROCEDURE РезервноеКопирование()  
BEGIN  
  DECLARE дамп_файл VARCHAR(100);  
  SET дамп_файл = CONCAT('backup_', DATE_FORMAT(NOW(),  
    '%Y%m%d_%H%i%s'), '.sql');
```

-- Использование утилиты mysqldump через системный вызов

```
SET @cmd = CONCAT('mysqldump -u root -p пароль стоматология >', дамп_файл);
```

-- В реальной среде выполнение осуществляется через системный вызов

```
END;
```

Форма выполнения: Отчет с описанием политик безопасности, скриптами резервного копирования, планом восстановления и настройками аудита.

4.1.6. Комплексное тестирование базы данных

Проведите комплексное тестирование разработанной базы данных:

1. Функциональное тестирование:
 - Разработайте тестовые сценарии для проверки всех объектов БД.
 - Проверьте корректность работы процедур, функций и триггеров.
 - Протестируйте граничные и некорректные значения.
2. Тестирование безопасности:
 - Проверьте права доступа пользователей.
 - Протестируйте механизмы шифрования.
 - Проверьте защиту от SQL-инъекций.
3. Нагрузочное тестирование:
 - Выполните тестирование с большим объемом данных.
 - Проверьте производительность сложных запросов.

Пример тестового сценария:

№	Тестовый сценарий	Ожидаемый результат	Фактический результат
1	Запись пациента на прием к врачу	Создается новая запись в таблице Приемы	Соответствует
2	Попытка записи на занятое время	Выводится сообщение "Врач занят"	Соответствует
3	Изменение статуса приема на "Выполнен"	Запись обновляется, триггер отрабатывает	Соответствует
4	Попытка доступа врача к данным другого врача	Доступ запрещен (ошибка прав)	Соответствует
5	Шифрование данных пациента	Данные шифруются при вставке/обновлении	Соответствует

Форма выполнения: Отчет о тестировании с описанием сценариев, результатов и выявленных ошибок.

4.2. Оценочные средства, применяемые для промежуточной аттестации по итогам изучения дисциплины

Зачет проводится по завершении производственной практики на последнем аудиторном занятии.

Промежуточная аттестация в форме зачета осуществляется по результатам сдачи отчета по практике и с учетом текущего контроля успеваемости при выполнении всех видов текущего контроля.

Обучающиеся, не выполнившие виды работ, предусмотренные рабочей программой; пропустившие более 50% практики без уважительной причины, не допускаются к зачету.

Промежуточная аттестация таких лиц проводится только после прохождения ими всех видов текущего контроля.

**V. ПОКАЗАТЕЛИ ОЦЕНКИ РЕЗУЛЬТАТОВ ОСВОЕНИЯ
ПРОИЗВОДСТВЕННОЙ ПРАКТИКИ**

Уровень сформированности компетенций	Оценка	Критерии оценивания по видам работ	
		тестирование (процент правильных ответов)	прочие виды работ по практике
Высокий	Отлично	90-100%	Обучающийся глубоко и прочно усвоил теоретический и освоил практический материал. Дает логичные и грамотные ответы. Демонстрирует знание не только основного, но и дополнительного материала, быстро ориентируется, отвечая на дополнительные вопросы. Свободно справляется с поставленными задачами, аргументировано и верно обосновывает принятые решения.
Повышенный	Хорошо	70-89%	Обучающийся твердо знает программный материал, грамотно и по существу излагает его. Не допускает существенных неточностей при ответах на вопросы, правильно применяет теоретические положения при решении практических задач, владеет навыками и приемами их выполнения.
Базовый	Удовлетворительно	50-69%	Обучающийся демонстрирует знания только основного материала, но не усвоил его детали, испытывает затруднения при решении практических задач. В ответах на поставленные вопросы допускает неточности. Дает определения понятий, искажающие их смысл. Нарушает последовательность изложения программного материала.
Не сформирована	Неудовлетворительно	0-49%	Обучающийся не знает, не выполняет или неправильно выполняет большую часть учебного материала. Допускает ошибки в формулировке определений, искажающие их смысл, беспорядочно и неуверенно излагает материал. Ответы на дополнительные вопросы отсутствуют. Не выполняет задания.

ЛИСТ РАССМОТРЕНИЙ И ОДОБРЕНИЙ

рабочей программы

ПП 01.01 Производственная практика (по профилю специальности)

в составе ООП 09.02.11 Разработка и управление программным обеспечением

1) <u>Рассмотрена и одобрена:</u>
а) На заседании предметно-цикловой методической комиссии протокол № 6 от 10.03.2026 г. Председатель ПЦМК  С.М. Нурбаева
б) На заседании методического совета протокол №4 от 21.04.2026 г. Председатель методического совета  М.В. Иваницкая
2) <u>Рассмотрена и одобрена внешним экспертом</u>
а) директор ООО «Сатори Партнер» А.Б. Мальцев