

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ
ПОВОЛЖСКИЙ ГОСУДАРСТВЕННЫЙ ТЕХНОЛОГИЧЕСКИЙ УНИВЕРСИТЕТ



УТВЕРЖДАЮ
Декан ЭФ

УТВЕРЖДАЮ /Н.М. Стрельникова/
(Ф.И.О. декана (директора института))

19.02.2026 г.

РАБОЧАЯ ПРОГРАММА ДИСЦИПЛИНЫ (МОДУЛЯ)

Б.1.1.20 Объектно-ориентированное программирование

(код и наименование дисциплины по учебному плану)

Направление подготовки
(специальность)

09.03.03 Прикладная информатика

Квалификация выпускника

Бакалавр

(бакалавр/магистр/специалист)

Направленность

Искусственный интеллект в бизнес-аналитике

Курс 2
Семестр 4

Распределение учебного времени

Трудоемкость по учебному плану	144 / 4	часов/зачетных единиц
Лекции	16	часов
Лабораторные работы	32	часов
Практические занятия	-	часов
Иная контактная работа	-	часов
Всего контактной работы (без учета экз.)	48	часов
Контактная работа по экзамену	-	часов
Курсовой проект (работа)	-	семестр
Самостоятельная работа обучающихся (без учета экз.)	96	часов
Самостоятельная работа по подготовке к экзамену	-	часов
Экзамен	-	семестр
Зачет	-	семестр
БРК, ДЗ	4	семестр

(год)

Программа составлена в соответствии с требованиями ФГОС ВО направления подготовки (специальности) 09.03.03 Прикладная информатика

Программу составили:

доцент	ИСЭ	СОГЛАСОВАНО	В.И. Абдулаев
(должность)	(кафедра)		(И.О. Фамилия)

РАССМОТРЕНА и ОДОБРЕНА на заседании кафедры, за которой закреплена дисциплина
Кафедра информационных систем в экономике

(наименование кафедры)		
17.02.2026	протокол №	10
(дата)		

Заведующий кафедрой	СОГЛАСОВАНО	Т.А. Уразаева
		(И.О. Фамилия)

Рабочая программа СОГЛАСОВАНА с факультетом (институтом), выпускающей(ими) кафедрой(ами).

СООТВЕТСТВУЕТ действующей ОП.

Заведующий кафедрой	СОГЛАСОВАНО	Т.А. Уразаева
		(И.О. Фамилия)

Председатель методической комиссии факультета (института), в который входит выпускающая кафедра

СОГЛАСОВАНО	О.Е. Иванов
	(И.О. Фамилия)

Эксперт(ы): Жубрин Алексей Анатольевич, помощник генерального директора ОАО «ММЗ» по информатизации – начальник управления информационных технологий

Рабочая программа проверена и зарегистрирована в УМЦ 19.02.2026 г.

Специалист учебно-методического центра СОГЛАСОВАНО /Т.А. Смирнова/

Раздел 1. ЦЕЛЬ ОСВОЕНИЯ ДИСЦИПЛИНЫ

Целью освоения дисциплины является достижение планируемых результатов обучения, соответствующих установленным в ОПОП индикаторам достижения компетенций:

Код и наименование компетенции	Код и наименование индикатора достижения компетенции	Результаты обучения
1. ОПК-2 Способен понимать принципы работы современных информационных технологий и программных средств, в том числе отечественного производства, и использовать их при решении задач профессиональной деятельности	ОПК-2.1. Знает современные информационные технологии и программные средства, в том числе отечественного производства при решении задач профессиональной деятельности	знания: ОПК-2.1. - Знать современные информационные технологии и программные средства, в том числе отечественного производства при решении задач профессиональной деятельности умения: навыки:
	ОПК-2.2. Умеет выбирать современные информационные технологии и программные средства, в том числе отечественного производства при решении задач профессиональной деятельности	знания: умения: ОПК-2.2. - Уметь выбирать современные информационные технологии и программные средства, в том числе отечественного производства при решении задач профессиональной деятельности навыки:
	ОПК-2.3. Владеет навыками применения современных информационных технологий и программных средств, в том числе отечественного производства, при решении задач профессиональной деятельности	знания: умения: навыки: ОПК-2.3. - Владеть навыками применения современных информационных технологий и программных средств, в том числе отечественного производства, при решении задач профессиональной деятельности

2. ОПК-3 Способен решать стандартные задачи профессиональной деятельности на основе информационной и библиографической культуры с применением информационно-коммуникационных технологий и с учетом основных требований информационной безопасности	ОПК-3.1. Знает принципы, методы и средства решения стандартных задач профессиональной деятельности на основе информационной и библиографической культуры с применением информационно-коммуникационных технологий и с учетом основных требований информационной безопасности	знания: ОПК-3.1. - Знать принципы, методы и средства решения стандартных задач профессиональной деятельности на основе информационной и библиографической культуры с применением информационно-коммуникационных технологий и с учетом основных требований информационной безопасности умения: навыки:
	ОПК-3.2. Умеет решать стандартные задачи профессиональной деятельности на основе информационной и библиографической культуры с применением информационно-коммуникационных технологий и с учетом основных требований информационной безопасности	знания: умения: ОПК-3.2. - Уметь решать стандартные задачи профессиональной деятельности на основе информационной и библиографической культуры с применением информационно-коммуникационных технологий и с учетом основных требований информационной безопасности навыки:
	ОПК-3.3. Владеет навыками подготовки обзоров, аннотаций, составления рефератов, научных докладов, публикаций, и библиографии по научно-исследовательской работе с учетом требований информационной безопасности	знания: умения: навыки: ОПК-3.3. - Владеть навыками подготовки обзоров, аннотаций, составления рефератов, научных докладов, публикаций, и библиографии по научно-исследовательской работе с учетом требований информационной безопасности

3. ОПК-7 Способен разрабатывать алгоритмы и программы, пригодные для практического применения	ОПК-7.1. Знает основные языки программирования и работы с базами данных, операционные системы и оболочки, современные программные среды разработки информационных систем и технологий	знания: ОПК-7.1. - Знать основные языки программирования и работы с базами данных, операционные системы и оболочки, современные программные среды разработки информационных систем и технологий умения: навыки:
	ОПК-7.2. Умеет применять языки программирования и работы с базами данных, современные программные среды разработки информационных систем и технологий для автоматизации бизнес-процессов, решения прикладных задач различных классов, ведения баз данных и информационных хранилищ	знания: умения: ОПК-7.2. - Уметь применять языки программирования и работы с базами данных, современные программные среды разработки информационных систем и технологий для автоматизации бизнес-процессов, решения прикладных задач различных классов, ведения баз данных и информационных хранилищ навыки:
	ОПК-7.3. Владеет навыками программирования, отладки и тестирования прототипов программно-технических	знания: умения: навыки: ОПК-7.3. - Владеть навыками программирования, отладки и тестирования прототипов программно-технических комплексов задач
4. PL-1 Способен применять язык программирования Python для решения задач в области ИИ	PL-1.1. Разрабатывает и отлаживает прикладные решения разной сложности и для разного круга конечных пользователей с использованием языка программирования Python, тестирует, испытывает и оценивает качество таких решений	знания: Б. Знает основы ООП, применяемые при разработке прикладных решений разной сложности и для разного круга конечных пользователей с использованием языка программирования Python, в том числе основы тестирования, и оценки качества таких решений. умения: Б. Умеет применять теорию ООП в контексте разработки программ для обработки и анализа данных навыки: Б. Владеет навыками создания небольших программ для обработки и анализа данных

PL-1.2. Осуществляет выбор инструментов разработки на Python, приемлимых для создания прикладной системы обработки научных данных, машинного обучения и визуализации с заданными требованиями	знания: Б. Знает наборы инструментов разработки на Питон, приемлемых для создания прикладной системы обработки данных умения: Б. Умеет подбирать инструменты для создания прикладной системы обработки данных навыки: Б. Владеет простыми инструментами обработки данных
PL-1.3. Разрабатывает и поддерживает системы обработки больших данных различной степени сложности	знания: Б. Знает архитектуру простых систем обработки больших данных умения: Б. Умеет поддерживать простые системы обработки больших данных навыки: Б. Владеет методами разработки простых систем обработки больших данных

Раздел 2. МЕСТО ДИСЦИПЛИНЫ В СТРУКТУРЕ ОПОП

Дисциплина относится к обязательной части ОПОП.

Дисциплина является обязательной

Для продолжения формирования заявленных компетенций необходимы знания предшествующих дисциплин: Информационные технологии (ОПК-2), Вычислительные системы, сети и телекоммуникации (ОПК-3), Алгоритмизация и программирование (ОПК-7), Алгоритмизация и программирование (PL-1)

Изучаемая дисциплина является основой для продолжения формирования указанных компетенций в следующих дисциплинах: Базы данных (ОПК-2), Информационная безопасность (ОПК-3), Программная инженерия (ОПК-7); практиках: Преддипломная практика (PL-1); государственной итоговой аттестации в форме: Выполнение и защита выпускной квалификационной работы (ОПК-2), Выполнение и защита выпускной квалификационной работы (ОПК-7), Выполнение и защита выпускной квалификационной работы (ОПК-3), Выполнение и защита выпускной квалификационной работы (PL-1)

Раздел 3. ОПИСАНИЕ ОБРАЗОВАТЕЛЬНЫХ ТЕХНОЛОГИЙ

Для формирования заявленных компетенций используются методологические технологии, реализующие деятельностный, личностно-ориентированный, практико-ориентированный подходы.

Основными стратегическими технологиями являются: исследовательские, лекционные занятия, практические и лабораторные занятия

На достижение конкретных целей обучения направлены применяемые тактические технологии: задания, классическая лекция, мини-проекты, лекция с обсуждением новых научных достижений в сфере ии по результатам конференции и публикаций , семинары с ит-практиками, конкурсы, хакатоны, кейс-чемпионаты, мастер-классы, командные проектные задания и др.

Раздел 4. СОДЕРЖАНИЕ ДИСЦИПЛИНЫ

Виды и темы занятий	Количество часов	Формируемые компетенции
Теория ООП и ее использование в системах бизнес аналитики	144	PL-1, ОПК-2, ОПК-3, ОПК-7
Лекция. Наследование и полиморфизм в конвейерах обработки.	2	
Лабораторная работа. Упражнения с иерархиями классов. `DataTransformer` (абстрактный класс с методом `transform(data)`) и его реализации `NormalizerTransformer` (нормализует числа) и `CategoryEncoderTransformer` (кодирует	2	
Лабораторная работа. Упражнения с функциями, основанными на полиморфизме.	2	
Лекция. Принципы SOLID на примерах.	2	
Лабораторная работа. Упражнения вида: Дан скрипт, который читает CSV, вычисляет статистику и рисует график. Разбить на 3 класса: `DataReader`, `StatisticsCalculator`, `PlotVisualizer`.	2	
Лабораторная работа. Упражнения с классами типа AnalysisPipeline, который в конструкторе принимает любой объект, с методом transform. Тестировать разные	2	
Лекция. Шаблоны проектирования: Стратегия и Наблюдатель	2	
Лабораторная работа. Упражнения на классы со стратегиями	2	
Лабораторная работа. Упражнения на классы с наблюдателями	2	
Лекция. Использование ООП для организации кода вокруг Pandas DataFrame.	2	
Лабораторная работа. Упражнения по обработке данных с помощью Pandas DataFrame.	4	
Лекция. Тестирование ООП кода. Основы pytest. Моки	2	
Лабораторная работа. Упражнения на тестирование классов с помощью pytest.	4	
Лекция. Оценка качества анализа данных. Метаданные и метрики	2	
Лабораторная работа. Создание системы валидации данных	2	
Лабораторная работа. Расчет метрик качества анализа	2	
Лекция. Типовые архитектурные решения для примера	2	
Лабораторная работа. Архитектурные решения для командных проектов.	4	
Лекция. Обзор курса. Подведение итогов проектного проектирования	2	
Лабораторная работа. Подведение итогов проектного проектирования	4	

Задания для самостоятельной работы, в том числе выполнение РГР Тема: "Система анализа продаж интернет-магазина" Задание: Создать набор классов, которые: 1. Читают данные из CSV-файлов (заказы, товары, клиенты). 2. Обработывают данные: объединяют таблицы, вычисляют выручку по заказу, добавляют новые признаки (например, "сезон" по дате заказа). 3. Анализируют данные: строят отчеты (топ-5 товаров по выручке, средний чек по месяцам). 4. Визуализируют результаты (строят один-два простых графика с помощью Matplotlib). Требования по ООП: Использовать инкапсуляцию (`@property` для вычисляемых метрик). Использовать наследование для создания разных типов отчетов (например, `Report` -> `TextReport`, `GraphReport`). Использовать принцип единственной ответственности (S) — отдельные классы для чтения, обработки, анализа и визуализации. Покрыть юнит-тестами ключевые методы (например, расчет выручки).	96
Иная контактная работа:	0

Раздел 5. МЕТОДИЧЕСКИЕ УКАЗАНИЯ ДЛЯ ОБУЧАЮЩИХСЯ ПО ОСВОЕНИЮ ДИСЦИПЛИНЫ

Изучение дисциплины рекомендуется начать с ознакомления с рабочей программой, ее структурой и содержанием разделов. Учебный материал структурирован, изучение дисциплины осуществляется в тематической последовательности. Занятия лекционного типа дают систематизированные знания по дисциплине, концентрируют внимание на наиболее сложных и важных вопросах. Во время лекционных занятий рекомендуется вести конспектирование учебного материала; обращать внимание на формулировки и категории, раскрывающие суть проблемы, явления или процесса; зафиксировать выводы и практические рекомендации. Подготовка к занятиям семинарского типа включает ознакомление с вопросами по темам занятия; работу с конспектом лекций, выполнение домашнего задания, работу с учебной и учебно-методической литературой, научными изданиями и электронными образовательными ресурсами, рекомендованными рабочей программой дисциплины.

Содержание самостоятельной работы определяется рабочей программой дисциплины, оценочными и методическими материалами, заданиями и указаниями преподавателя. Самостоятельная работа может осуществляться в аудиторной и внеаудиторной формах. Обеспечено разнообразие форматов и методов обучения (лекции, семинары с ИТ-практиками, конкурсы, хакатоны, кейс-чемпионаты, мастер-классы, командные проектные задания и др.) в каждом семестре. Формой промежуточной аттестации по дисциплине являются БРК.

Эффективным средством осуществления самостоятельной работы является электронная информационно-образовательная среда университета, которая обеспечивает доступ к образовательной программе, рабочей программе дисциплины, к электронным библиотечным системам, профессиональным базам данных и информационным справочным системам. К

самостоятельному изучению рекомендуется следующая литература:

- 1) Лутц М. Изучаем Python. — СПб.: Диалектика, 2019. — 832 с. — ISBN 978-5-907144-85-0
- 2) Любанович Б. Простой Python. Современный стиль программирования. — СПб.: Питер, 2021. — 592 с. — ISBN 978-5-4461-1456-6
- 3) Шоу Э. Внутри CPython: гид по интерпретатору Python. — СПб.: Питер, 2023. — 352 с. — ISBN 978-5-4461-1925-7
- 4) Жерон О. Прикладное машинное обучение с помощью Scikit-Learn и TensorFlow / пер. А. А. Слинкина. — М.: ДМК, 2023. — 688 с. — ISBN 978-5-93700-123-7
- 5) Рашка С. Машинное обучение с PyTorch и Scikit-Learn. — М.: ДМК, 2023. — 790 с. — ISBN 978-5-93700-145-9
- 6) ПНСТ 838-2023 Искусственный интеллект. Структура описания систем искусственного интеллекта, использующих машинное обучение
- 7) ГОСТ Р 71752-2024 Искусственный интеллект. Техническое задание. Требования к содержанию
- 8) Сопов Е. А. Технологии обработки больших данных на Python: учебник. — М.: ИНФРА-М, 2023. — 156 с. — ISBN 978-5-16-019627-5
- 9) Силен Д., Мейсман А., Али М. Основы Data Science и Big Data. Python и наука о данных. — СПб.: Питер, 2017. — 336 с. — ISBN 978-5-496-02545-3
- 10) МакГрегор С. Обработка данных на Python. Data Wrangling и Data Quality. — БХВ, 2024. — 320 с. — ISBN 978-5-9775-1846-8
- 11) Марин И., Шукла А., ВК С. Big Data Analysis with Python: Combine Spark and Python to unlock the powers of parallel computing and machine learning. — Packt Publishing, 2019. — 276 с. — ISBN 978-1789955286
- 12) Мюллер А., Гвидо С. Введение в машинное обучение с помощью Python. — М.: ДМК, 2019. — 480 с. — ISBN 978-5-97060-660-5

Предполагается также изучение исследований и разработок, представленных в научных публикациях на конференциях уровня А*, статьях, опубликованных в журналах 1 уровня Белого списка научных журналов Минобрнауки России (УММ на основе публикаций в научных журналах 1 или 2 уровня Белого списка научных журналов Минобрнауки России, например:

1. СИБИРСКИЙ МАТЕМАТИЧЕСКИЙ ЖУРНАЛ
2. СИБИРСКИЙ ЖУРНАЛ ВЫЧИСЛИТЕЛЬНОЙ МАТЕМАТИКИ
3. ДИСКРЕТНАЯ МАТЕМАТИКА
4. ДИСКРЕТНЫЙ АНАЛИЗ И ИССЛЕДОВАНИЕ ОПЕРАЦИЙ
5. ВЫЧИСЛИТЕЛЬНЫЕ МЕТОДЫ И ПРОГРАММИРОВАНИЕ
6. ИСКУССТВЕННЫЙ ИНТЕЛЛЕКТ И ПРИНЯТИЕ РЕШЕНИЙ
7. ИНФОРМАЦИОННЫЕ ПРОЦЕССЫ
8. VESTNIK ROSSIYSKIH UNIVERSITETOV. МАТЕМАТИКА|ВЕСТНИК РОССИЙСКИХ УНИВЕРСИТЕТОВ. МАТЕМАТИКА,

, публикаций на конференциях уровня А*/А), например:

ACM Conference on Economics and Computation;

European Conference on Artificial Intelligence

International Conference on Learning Analytics and Knowledge

Раздел 6. МАТЕРИАЛЬНО-ТЕХНИЧЕСКОЕ И УЧЕБНО-МЕТОДИЧЕСКОЕ ОБЕСПЕЧЕНИЕ ДИСЦИПЛИНЫ

6.1. Учебно-методическое обеспечение

№№ п/п	Список используемой литературы	Количество экземпляров печатных изданий, имеющихся в библиотеке, или электронный адрес издания (ресурса) в сети Интернет
УЧЕБНЫЕ, УЧЕБНО-МЕТОДИЧЕСКИЕ И НАУЧНЫЕ ИЗДАНИЯ		
1.	Федоров, Дмитрий Юрьевич. Программирование на языке высокого уровня Python [Текст : Электронный ресурс] : учебное пособие для вузов / Д. Ю. Федоров. 5-е изд. Москва: Юрайт, 2025. - 210 с ISBN 978-5-534-14638-7.	https://urait.ru/book/programirovanie-na-python-556864
2.	Sweigart, А. Разработка компьютерных игр на языке Python [Электронный ресурс] / Sweigart А. 2-е изд. Москва: ИНТУИТ, 2016. - 504 с.	https://e.lanbook.com/book/100454
3.	Sweigart, А. Разработка компьютерных игр с помощью Python и Pygame [Электронный ресурс] / Sweigart А. 2-е изд. Москва: ИНТУИТ, 2016. - 289 с.	https://e.lanbook.com/book/100455
4.	Букунов, С. В. Разработка приложений с графическим пользовательским интерфейсом на языке Python [Электронный ресурс] : учебное пособие для вузов / Букунов С. В., Букунова О. В.; Букунова О. В. Санкт-Петербург: Лань, 2023. - 88 с. ISBN 978-5-507-45191-3.	https://e.lanbook.com/book/292856
ЭЛЕКТРОННЫЕ ОБРАЗОВАТЕЛЬНЫЕ РЕСУРСЫ		
1.	ОСНОВЫ ПРОГРАММИРОВАНИЯ НА ЯЗЫКЕ PYTHON	https://www.elibrary.ru/item.asp?id=50449034
2.	ИСПОЛЬЗОВАНИЕ ПРОГРАММ ГЕНЕРАТИВНОГО ИСКУССТВЕННОГО ИНТЕЛЛЕКТА В ПРОЦЕССЕ НАЧАЛЬНОГО ОБУЧЕНИЯ ПРОГРАММИРОВАНИЮ В ВУЗЕ	https://www.elibrary.ru/item.asp?id=62498462
3.	PYTHON для анализа больших данных	https://www.elibrary.ru/item.asp?id=80605670

6.2. Материально-техническая база и программное обеспечение

№№ п/п	Аудитории для проведения учебных занятий, самостоятельной работы и проведения государственной итоговой аттестации	Перечень основного оборудования	Программное обеспечение
1.	314а (III)	ПК 1 - ICL RAY P222.1 ,клавиат.,мышь.,монитор NEC 23" LCD EX 231WP-BK (1), ПК 2 - ICL RAY P222.2 ,клавиат.,мышь.,монитор NEC 23" LCD EX 231WP-BK (15), Проектор мультимедийный Hitachi	Microsoft Windows Enterprise, Справочная правовая система "Консультант Плюс", Microsoft Office Standard, Агент Dr.Web, Комплект ГАРАНТ-

		CP-RX93 (1), Шкаф IBM Netbay 25U (1), Экран настенный рулонный 200x200 см (1), Комплект учебной мебели (1)	Мастер, Microsoft Access, Microsoft Visio Professional, Microsoft Project Professional, Microsoft Visual Studio Enterprise, Комплект ПО для решения основных пользовательских задач
--	--	--	---

Раздел 7. ФОРМЫ КОНТРОЛЯ ОСВОЕНИЯ ДИСЦИПЛИНЫ/ ФОНД ОЦЕНОЧНЫХ СРЕДСТВ

Критерии оценивания индикаторов достижения компетенций направлены на:

- усвоение теоретического материала (объем знаний, глубина усвоения), предусмотренного рабочей программой;
- умение излагать материал (четкость, грамотность изложения материала, точность и полнота воспроизведения учебного материала);
- умение применять теоретические знания при решении практических заданий.

Шкала оценивания представлена ниже.

Уровень сформированности элементов компетенции	Критерии оценивания	Шкала оценивания
Пороговый уровень	Обучающийся имеет знания основного материала, проявляет умение логично его излагать, но может допускать неточности в изложении материала, недостаточно правильные формулировки, испытывает затруднения в выполнении практических заданий.	удовлетворительно
Продвинутый уровень	Обучающийся твердо знает программный материал, излагает его грамотно и по существу, не допускает существенных неточностей в ответе на вопрос, правильно применяет теоретические положения при решении практических вопросов и задач, владеет необходимыми навыками и приемами их выполнения	хорошо
Высокий уровень	Обучающийся глубоко и прочно усвоил программный материал, грамотно и логически стройно его излагает, дает исчерпывающие ответы на поставленные вопросы. В ответе тесно увязывается теория с практикой, при этом обучающийся не затрудняется с ответом при видоизменении задания, свободно справляется с задачами, вопросами и другими видами применения знаний, показывает знакомство с монографической литературой, периодическими изданиями, правильно обосновывает принятые решения, свободно владеет разносторонними навыками, приемами выполнения практических работ	отлично

7.1. Текущий контроль успеваемости

Текущий контроль успеваемости обеспечивает оценивание хода освоения дисциплины (модуля) и производится с применением технологии рейтингового контроля в соответствии с технологической картой дисциплины. Порядок составления технологической карты и алгоритм проведения процедуры оценивания видов деятельности обучающихся,

направленных на освоение знаний, умений, навыков и/ или опыта деятельности, по накопительной системе в баллах устанавливается положением о системе РИТМ в ФГБОУ ВО «ПГТУ»

7.2. Промежуточная аттестация обучающихся

Промежуточная аттестация обучающихся направлена на оценивание результатов обучения по дисциплине (модулю) и проводится с использованием фондов оценочных средств.

Примеры типовых контрольных заданий из базы фонда оценочных средств по образовательной программе.

Примеры типовых контрольных заданий (проектов) из базы фонда оценочных средств по образовательной программе

1. Разработать модульную систему для загрузки, обработки и анализа данных о продажах интернет магазина, используя следующие классы и методы:

Цель: Разработать модульную систему для загрузки, обработки и анализа данных о продажах.

Сущности и классы:

1). ``DataManager`` (абстрактный базовый класс)

Наследники: ``CSVDataManager``, ``JSONDataManager``

Методы: ``load_data()``, ``save_data()``

2). ``SalesAnalyzer``

Методы: ``get_total_revenue()``, ``get_best_selling_products()``, ``get_monthly_trends()``

3). ``ReportGenerator`` (абстрактный базовый класс)

Наследники: ``TextReportGenerator``, ``VisualReportGenerator`` (строит графики через matplotlib)

Метод: ``generate_report(analyzer)``

4). ``DataValidator``

Методы: ``validate_schema()``, ``validate_negative_prices()``, ``validate_missing_dates()``

Использует ``@property`` для проверки данных при установке значений

2. Система управления библиотечным фондом с аналитикой

Цель: Создать систему для учета книг в библиотеке с базовой аналитикой популярности и статистикой.

Сущности и классы:

Book

Атрибуты: id, title, author, year, genre, is_available

Методы: ``__str__``, ``__eq__`` (для сравнения по id)

Инкапсуляция: Сеттер для is_available проверяет, что значение булевое

Library

Атрибут: books (список объектов Book)

Методы:

``add_book(book)``, ``remove_book(book_id)``

find_books_by_author(author), find_books_by_genre(genre)
borrow_book(book_id), return_book(book_id) (меняют is_available)
LibraryAnalyzer
Принимает в конструктор объект Library
Методы:
get_most_popular_genre()
get_books_by_decade(decade)
get_availability_statistics() (% доступных книг)
LibraryReport
Методы: generate_collection_report(analyzer), generate_availability_report(analyzer)

Перечень вопросов для проведения промежуточной аттестации

Вопросы к БРК по дисциплине «Объектно-ориентированное программирование»

1. В чем разница между процедурным и объектно-ориентированным подходами при обработке данных? Какой из них лучше для сложного анализа и почему?
2. Объясните, для чего нужны "приватные" (`\_` и `\_\_`) атрибуты в Python. Приведите пример из области анализа данных, где их использование критически важно.
3. Что такое геттеры и сеттеры "в стиле Python" (`@property`)? Как с их помощью можно обеспечить валидацию данных при создании объекта, например, класса `Customer`?
4. Чем полезны магические методы `\_\_eq\_\_` и `\_\_str\_\_` при работе с наборами данных? Приведите пример использования каждого.
5. Спроектируйте класс `DataRecord` для хранения одной строки данных, инкапсулирующий сырые значения и предоставляющий свойства для доступа к предобработанным данным.
6. В чем заключается принцип подстановки Барбары Лисков (LSP)? Приведите пример корректного и некорректного наследования в иерархии классов для трансформации данных.
7. Что такое абстрактный базовый класс (ABC) и какой проблемой он помогает решить при проектировании конвейеров обработки данных?
8. Объясните на примере, как полиморфизм позволяет создавать расширяемые конвейеры обработки данных. Что произойдет, если мы добавим новый класс-трансформер?
9. В чем ключевое различие между наследованием ("is-a") и композицией ("has-a")? Какой подход предпочтительнее для построения гибких ETL-конвейеров и почему?
10. Спроектируйте иерархию классов для работы с различными источниками данных (CSV, JSON, база данных), используя наследование от абстрактного класса `DataSource`.
11. Что означает Принцип единственной ответственности (SRP) и как его нарушение в классе `DataProcessor` может усложнить его поддержку и тестирование?
12. Проиллюстрируйте на примере Принцип открытости/закрытости (OCP). Как он помогает добавлять новые алгоритмы агрегации данных, не меняя существующий код?
13. Объясните, как Принцип инверсии зависимостей (DIP) делает код анализа данных менее связанным. Что такое "зависимость от абстракций, а не от деталей"?
14. Класс `ReportGenerator` зависит от конкретного класса `MySQLConnector`. Как нарушается DIP и

как это можно исправить?

15. Проанализируйте предложенный код класса и укажите, какие принципы SOLID в нем нарушены и как их можно исправить.
16. Опишите проблему, которую решает шаблон "Стратегия". Приведите пример из анализа данных, где его применение было бы уместно.
17. Как шаблон "Стратегия" связан с Принципом открытости/закрытости (OCP) из SOLID?
18. В чем заключается идея шаблона "Наблюдатель"? Как его можно использовать для мониторинга выполнения длительных задач обработки данных (например, обучения модели)?
19. Спроектируйте систему, где конвейер обработки данных уведомляет логгер и панель мониторинга о своем прогрессе, используя шаблон "Наблюдатель".
20. В чем разница между Шаблоном "Стратегия" и наследованием? Когда лучше использовать каждый из подходов?
21. Каковы преимущества создания класса-обертки вокруг pandas DataFrame? Приведите два конкретных примера такой обертки для предметной области.
22. Как с помощью ООП можно обеспечить целостность и валидность данных при их добавлении в DataFrame? Проиллюстрируйте на примере метода `add_sale()`.
23. Опишите, как можно использовать композицию для построения класса `DataAnalyzer`, который инкапсулирует внутри себя DataFrame и методы для анализа.
24. В чем опасность создания "божественного класса" (God Class), который знает и умеет слишком много в контексте анализа данных? К каким последствиям это может привести?
25. Спроектируйте класс `TimeSeriesData` для работы с временными рядами, который будет расширять функциональность pandas DataFrame добавлением специфичных методов (например, `remove_trend()`, `calculate_seasonality()`).
26. Почему тестирование особенно важно для кода, связанного с обработкой и анализом данных? Приведите две причины.
27. Что такое "мок" (mock) и зачем он нужен в юнит-тестировании класса, который загружает данные из внешнего API?
28. Как с помощью фикстур в `pytest` можно избежать дублирования кода при создании тестовых данных для нескольких тестовых методов?
29. Напишите структуру юнит-теста для метода `calculate_customer_lifetime_value()` класса `CustomerAnalytics`, учитывая подготовку тестовых данных и проверку ожидаемого результата.
30. Протестируйте класс `DataCleaner`, который имеет зависимость от внешнего сервиса. Как обеспечить изоляцию теста и проверить, что класс корректно обрабатывает ошибки от этого сервиса?

Лекция 7: Оценка качества анализа данных. Метаданные и метрики

31. Что подразумевается под "качеством анализа данных"? Назовите и охарактеризуйте три различных аспекта такого качества.
32. Как ООП помогает систематизировать расчет и отслеживание метрик качества данных и качества анализа? Какие классы для этого можно создать?

33. Опишите, как можно использовать шаблон "Компоновщик" (Composite) для построения системы сбора разнородных метрик качества (метрики данных, скорости, бизнес-метрики).
34. Спроектируйте класс `DataValidator`, который может последовательно применять несколько проверок (валидация схемы, диапазонов, полноты) к набору данных и формировать отчет об ошибках.
35. Почему принцип "Мусор на входе — мусор на выходе" (Garbage In, Garbage Out) особенно важен для автоматизированных систем анализа данных? Какие инструменты ООП помогают бороться с этой проблемой?