



**ФЕДЕРАЛЬНОЕ АГЕНТСТВО ВОЗДУШНОГО ТРАНСПОРТА
(РОСАВИАЦИЯ)**

**ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ
ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ
«САНКТ-ПЕТЕРБУРГСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ
ГРАЖДАНСКОЙ АВИАЦИИ
ИМЕНИ ГЛАВНОГО МАРШАЛА АВИАЦИИ А.А. НОВИКОВА»**

УТВЕРЖДАЮ

Ректор

- / Ю. Ю. Михальчевский

«__23__» апреля 2026 года

РАБОЧАЯ ПРОГРАММА ДИСЦИПЛИНЫ

Информатика

Направление подготовки:

23.03.01 Технология транспортных процессов

Профиль

Транспортная логистика и цифровые технологии

Квалификация выпускника

бакалавр

Форма обучения

заочная

Санкт-Петербург

2026

1 Цели освоения дисциплины

Цель освоения дисциплины «Информатика» – получение теоретических сведений об информатике, получение теоретических сведений о способах хранения, представления и обработки информации, получение практических навыков решения широкого круга задач с использованием персонального компьютера, развитие самостоятельности при решении задач с использованием открытых источников информации.

2 Место дисциплины в структуре ОПОП ВО

Дисциплина «Информатика» представляет собой дисциплину, относящуюся к Обязательной части Блока 1 дисциплин ОПОП ВО по направлению подготовки 23.03.01 «Технология транспортных процессов» (бакалавриат), профиль «Транспортная логистика и цифровые технологии».

Дисциплина «Информатика» является обеспечивающей для следующих дисциплин: «Цифровая логистика на транспорте», «Интернет технологии и цифровая безопасность на транспорте», «Базы и банки данных на транспорте», «Статистика и анализ данных на транспорте», «Системы искусственного интеллекта на транспорте».

Дисциплина изучается во 2 семестре.

3 Компетенции обучающегося, формируемые в результате освоения дисциплины

Процесс освоения дисциплины направлен на формирование следующих компетенций:

Код компетенции/ индикатора	Результат обучения: наименование компетенции, индикатора компетенции
ОПК-4	Способен понимать принципы работы современных информационных технологий и использовать их для решения задач профессиональной деятельности.
ИД ¹ _{ОПК4}	Ориентируется в пакетах прикладных программ, работает с программными средствами, применяет современные информационные технологии.
ИД ² _{ОПК4}	Выбирает и использует современные информационные технологии и программные средства для решения поставленных задач, в том числе в сфере профессиональной деятельности.

Планируемые результаты изучения дисциплины:

Знать:

- основные принципы самоорганизации и самообразования;
- основные сведения о дискретных структурах, используемых в персональных компьютерах;
- способы обработки материала с применением современных информационных технологий;
- способы сбора библиографической информации по дисциплине;

-структуру локальных и глобальных компьютерных сетей.

Уметь:

- воспринимать и реализовывать на практике полученные знания;
- работать в качестве пользователя персонального компьютера;
- самостоятельно применять всю совокупность полученных знаний;
- обрабатывать и анализировать материал с применением современных информационных технологий;
- работать с компьютером как средством управления информацией.

Владеть:

- методами сбора, хранения и обработки информации, применяемыми в профессиональной деятельности;
- методами самоорганизации и самообразования;
- методами обработки материала, знаниями по информационной безопасности;
- методами решения функциональных и вычислительных задач.

4 Объём дисциплины и виды учебной работы

Общая трудоемкость дисциплины составляет 4 зачетные единицы, 144 академических часа.

Наименование	Всего часов	Семестр 2
Общая трудоёмкость дисциплины(модуля)	144	144
Контактная работа:	6,5	6,5
Лекции	2	2
Практические занятия	4	4
Лабораторные работы		
Самостоятельная работа студента	129	129
Промежуточная аттестация (экзамен), в т. ч.	9	9
контактная работа	2,5	2,5
самостоятельная работа по подготовке к экзамену	6,5	6,5

5. Содержание дисциплины

5.1 Соотнесение тем дисциплины и формируемых компетенций.

Темы дисциплины	Количество часов	Компетенции	Образовательные технологии	Оценочные средства
		ОПК-4		
Тема 1. Основы алгоритмизации и язык программирования Python	48	+	Л, ПЗ, СРС	ИЗ, У
Тема 2. Анализ и визуализация данных	37	+	Л, ПЗ, СРС	ИЗ, У
Тема 3. Библиотека Pandas.	23	+	Л, ПЗ, СРС	ИЗ, У
Итого по дисциплине	108			
Промежуточная аттестация	36			
Всего по дисциплине	144			

Л– лекция, ПЗ – практическое занятие, СРС – самостоятельная работа, У– устный опрос, ИЗ - индивидуальное задание.

5.2 Темы дисциплины и виды занятий

Наименование темы дисциплины	Л	ПЗ	КР	СРС	ЛР	Всего часов
Тема 1. Основы алгоритмизации и язык программирования Python	1	2	-	59	-	62
Тема 2. Анализ и визуализация данных	0,5	1	-	35	-	36,5
Тема 3. Библиотека Pandas.	0,5	1	-	35	-	36,5
Итого по дисциплине	2	4		129		135
Промежуточная аттестация						9
Всего по дисциплине						144

Л – лекция, ПЗ – практические занятия, КР – контрольная работа, ЛР – лабораторная работа, СРС – самостоятельная работа.

5.3 Содержание тем дисциплины

Тема 1. Основы алгоритмизации и программирования Python

Понятие алгоритма и его роль в автоматизации логистических процессов. Визуализация алгоритмов: блок-схемы как инструмент описания последовательности действий. Примеры логистических задач, решаемых с помощью алгоритмов. Обзор языка Python и среды разработки (IDE, Jupyter Notebook). Структура первой программы. Функция print() для вывода информации. Переменные как именованные ячейки памяти. Основные типы данных: строки и числа. Функция input() для получения данных от пользователя. Арифметические операторы (+, -, *, /) для выполнения расчётов. Демонстрация построения простой формулы для расчёта стоимости перевозки. Логика ветвления: операторы if, else, elif. Операторы сравнения для проверки условий. Примеры программ,

принимающих решения на основе входных данных. Цикл `for` для перебора последовательностей (списков). Функция `range()` для генерации числовых последовательностей. Демонстрация обработки списка грузов или городов с помощью цикла. Списки (list) как упорядоченные коллекции элементов. Словари (dict) как структуры «ключ-значение» для хранения справочных данных. Понятие функции как именованного блока кода. Создание собственных функций для инкапсуляции логики расчётов. Передача аргументов в функцию и возврат результата.

Тема 2. Анализ и визуализация данных

Наука о данных. Структуры данных. Данные, наука о данных, открытые данные, источники данных, структуры данных (стек, массив, очередь, хэш-таблица). Работа с таблицами и подготовка данных. Базовые статистические и логические функции, функции прогнозирования в табличном процессоре, поиск, очистка, преобразование, организация и сбор данных. Роль визуализации в анализе логистических показателей. Обзор библиотеки `matplotlib` для построения графиков.

Тема 3. Библиотека Pandas: таблицы в Python

Библиотеки Python. Библиотека Pandas. Поиск, очистка, преобразование, организация и сбор данных. Библиотека языка программирования, библиотеки Python, библиотека Pandas, импорт библиотек. Структуры данных в Pandas. Поиск, очистка, преобразование, организация и сбор данных, структуры данных в Pandas. Структура данных Series. Структура данных DataFrame. Структура данных DataFrame, словарь, список, функция `read_csv`, методы `head` и `tail`. Информация о данных, методы `info` и `describe`, числовые и категориальные признаки. Агрегирующие функции `value_counts`, `unique`, `nunique`, `groupby`. Методы `min()`, `max()` и `mean()`. Объединение таблиц с помощью метода `merge`, параметры `on` и `how`. Библиотека визуализации данных. Визуализация данных, преимущества диаграмм и графиков. Виды диаграмм. Библиотеки Pandas, Matplotlib, Seaborn и построение графиков и диаграмм с помощью этих библиотек. Библиотека NumPy. Библиотека NumPy, массив. Массив в NumPy, характеристики массивов, их отличие от известных ранее структур данных, размерность массива, тип данных, доступ к элементам массива. Работа с массивами NumPy: создание, вывод элементов массива, операции над массивами.

5.4 Практические занятия

Номер темы дисциплины	Тематика практических занятий	Трудоемкость (часы)
1	Практическое занятие №1. Базовый синтаксис, структуры данных и алгоритмы.	1
1	Практическое занятие №2. Файлы, обработка ошибок, ООП.	1

2	Практическое занятие №3. Анализ данных в табличном процессоре.	0,5
2	Практическое занятие №4. Визуализация в анализе данных (построение дашбордов).	0,5
3	Практическое занятие №5. Анализ данных с использованием библиотеки Pandas	1
Итого по дисциплине		4

Краткое содержание лабораторных работ в 2-м семестре:

Практическое занятие №1. Базовый синтаксис, структуры данных и алгоритмы. Объявление переменных и типы данных. Условные конструкции: if, elif, else. Циклы: for и while. Понятие отступов и их роль в Python. Создание и вызов функций, работа с аргументами и возвращаемыми значениями. Области видимости переменных (local, global). Основные встроенные структуры: список (list), кортеж (tuple), множество (set), словарь (dict). Операции с элементами: добавление, удаление, изменение. Срезы (slices) для списков и строк. Преобразование и объединение структур данных. Списковые включения (list comprehensions). Методы словарей: get, keys, values, items. Линейный и бинарный поиск в списке. Реализация базовых алгоритмов сортировки: пузырьком, выбором, вставками. Использование встроенных функций сортировки. Поиск максимального, минимального, среднего значения. Удаление дубликатов из списка. Простые алгоритмы для работы со строками.

Практическое занятие №2. Файлы, обработка ошибок, ООП. Открытие, чтение и запись файлов: функции open, read, write, close. Использование контекстного менеджера with для работы с файлами. Работа с разными режимами открытия файлов (r, w, a, b и др.). Обработка текстовых и бинарных файлов. Конструкция try/except/else/finally: синтаксис и примеры использования. Обработка стандартных исключений (FileNotFoundError, ValueError, TypeError и др.) Генерация собственных исключений (raise). Логирование ошибок с помощью модуля logging. Создание классов, атрибутов и методов. Инкапсуляция: уровни доступа к атрибутам (public, protected, private). Конструктор класса: метод init. Наследование. Полиморфизм.

Практическое занятие №3. Анализ данных в табличном процессоре. Импорт данных из различных источников (CSV, TXT, базы данных). Преобразование форматов данных: текст, числа, даты. Сортировка и фильтрация таблиц. Математические и статистические функции (СУММ, СРЗНАЧ, МАКС, МИН, СТАНДОТКЛОН); условные функции (ЕСЛИ, СУММЕСЛИ, СЧЁТЕСЛИ, СУММЕСЛИМН); функции поиска и извлечения данных (ВПР, ГПР, ИНДЕКС, ПОИСКПОЗ). Создание и настройка сводных таблиц: группировка, вычисляемые поля, детализация. Использование срезов и временных шкал для интерактивной фильтрации. Построение сводных диаграмм и графиков для визуализации результатов анализа. Использование функций прогнозирования (ТЕНДЕНЦИЯ,

ПРЕДСКАЗ, ПРОГНОЗ.ETS). Анализ сезонности и построение прогнозов на основе исторических данных.

Практическое занятие №4. Визуализация в анализе данных (построение дашбордов). Загрузка и подключение файла с данными (например, CSV). Формирование датасета на основе подключённого файла. Определение типов полей: показатели и измерения. Создание вычисляемых полей (например, средний вес, количество уникальных значений). Создание первого чарта: столбчатая диаграмма для анализа распределения по категориям. Настройка осей, сортировки, фильтров и цветовых схем. Сохранение и настройка визуальных параметров чартов. Добавление на дашборд ранее созданных чартов. Настройка расположения и размеров виджетов. Добавление селекторов (фильтров) для анализа по различным признакам. Настройка множественного выбора и зависимостей между элементами дашборда. Применение фильтров и анализ зависимостей между показателями. Исследование данных: выявление трендов, аномалий, сравнительный анализ групп. Построение геоаналитических карт.

Практическое занятие №5. Анализ данных с использованием библиотеки Pandas. Загрузить набор данных и провести его предварительный анализ. Очистить данные: обработать пропуски, привести типы к нужному формату. Сформировать новые показатели (например, рассчитать средний чек, прирост по месяцам). Построить сводную таблицу для анализа структуры продаж по категориям и регионам. Объединить несколько таблиц в одну для комплексного анализа. Построение базовых графиков (line, bar, hist) на основе DataFrame. Использование встроенных методов визуализации для экспресс-анализа

5.5 Лабораторный практикум

Проведение лабораторных занятий учебным планом не предусмотрено.

5.6 Самостоятельная работа

Номер темы дисциплины	Виды самостоятельной работы	Трудоемкость (часы)
1	Изучение теоретического материала. Подготовка к устному опросу.	22
2	Изучение теоретического материала. Подготовка к устному опросу	17
3	Изучение теоретического материала. Подготовка к устному опросу.	15
Всего за 2 семестр		54
Итого по дисциплине		54

5.7 Курсовые работы

Курсовые работы (проекты) учебным планом не предусмотрены.

6 Учебно-методическое и информационное обеспечение дисциплины

а) основная литература:

1. Операционная система «РЕД ОС». Руководство пользователя. RU.29926343.02.01-01 34 1-1 <https://redos.red-soft.ru/product/docs/?ysclid=m98n9lywz2379970398> (дата обращения 08. 04. 2025)

2. Astra Linux. Руководство по национальной операционной системе и совместимым офисным программам <https://astra.ru/info/reference-information/library/> (дата обращения 08. 04. 2025)

3. Черпаков, И. В. Алгоритмизация и программирование на Python : учебник для вузов / И. В. Черпаков. — Москва : Издательство Юрайт, 2026. — 159 с. — (Высшее образование). — ISBN 978-5-534-21910-4. — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: <https://urait.ru/bcode/582412> (дата обращения: 10.04.2026).

4. Воронов, М. В. Системы искусственного интеллекта : учебник и практикум для вузов / М. В. Воронов, В. И. Пименов, И. А. Небаев. — 2-е изд., перераб. и доп. — Москва : Издательство Юрайт, 2026. — 268 с. — (Высшее образование). — ISBN 978-5-534-17032-0. — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: <https://urait.ru/bcode/588642> (дата обращения: 15.04.2026).

5. Чернышев, С. А. Основы программирования на Python : учебник для вузов / С. А. Чернышев. — 2-е изд., перераб. и доп. — Москва : Издательство Юрайт, 2026. — 349 с. — (Высшее образование). — ISBN 978-5-534-17139-6. — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: <https://urait.ru/bcode/588667> (дата обращения: 15.04.2026).

б) дополнительная литература:

6. Электронно-библиотечная система издательства «Лань» [Электронный ресурс] – Режим доступа: <http://e.lanbook.com/>, свободный (дата обращения: 8.04.2025);

7. Анализ данных : учебник для вузов / под редакцией В. С. Мхитаряна. — Москва : Издательство Юрайт, 2026. — 448 с. — (Высшее образование). — ISBN 978-5-534-19964-2. — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: <https://urait.ru/bcode/583032> (дата обращения: 15.04.2026).

8. Платонов, А. В. Машинное обучение : учебное пособие для вузов / А. В. Платонов. — 2-е изд. — Москва : Издательство Юрайт, 2026. — 89 с. — (Высшее образование). — ISBN 978-5-534-20732-3. — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: <https://urait.ru/bcode/589132> (дата обращения: 15.04.2026).

9. Федоров, Д. Ю. Программирование на Python : учебное пособие для вузов / Д. Ю. Федоров. — 6-е изд., перераб. и доп. — Москва : Издательство Юрайт, 2026. — 187 с. — (Высшее образование). — ISBN 978-5-534-19666-5. — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: <https://urait.ru/bcode/585806> (дата обращения: 15.04.2026).

10. Воронов, М. В. Системы искусственного интеллекта : учебник и практикум для вузов / М. В. Воронов, В. И. Пименов, И. А. Небаев. — 2-е изд., перераб. и доп. — Москва : Издательство Юрайт, 2026. — 268 с. — (Высшее

образование). — ISBN 978-5-534-17032-0. — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: <https://urait.ru/bcode/588642> (дата обращения: 15.04.2026).

7 Материально-техническое обеспечение дисциплины

1 Компьютерный класс, оборудованный ПК индивидуально для каждого студента. (ауд. 801-808)

2 Инсталлированные изучаемые средства прикладного и инструментального ПО: РЕД ОС, Яндекс Браузер, LibreOffice Writer, LibreOffice Calc, IDLE, Python (с библиотеками Pandas, NumPy, Matplotlib, Scikit-learn), Jupyter Notebook.

3 Проектор при чтении лекции.

4 Интерактивная доска для проведения практических занятий.

8 Образовательные технологии

В рамках изучения дисциплины предполагается использовать следующие образовательные технологии: лекции, практические занятия, самостоятельная работа студента.

Лекции. Традиционная лекция составляет основу теоретического обучения в рамках дисциплины и направлена на систематизированное изложение накопленных и актуальных научных знаний. На лекции концентрируется внимание обучающихся на наиболее сложных и узловых вопросах, стимулируется их активная познавательная деятельность. Ведущим методом в лекции выступает устное изложение учебного материала.

Практические занятия по дисциплине проводятся в соответствии с учебно-тематическим планом. Цель практических занятий – закрепить теоретические знания, полученные обучающимися на лекциях и в результате самостоятельного изучения соответствующих разделов рекомендуемой литературы, а также приобрести начальные практические навыки.

Практические занятия предназначены для более глубокого освоения и анализа тем, изучаемых в рамках данной дисциплины.

Самостоятельная работа студентов включает:

- а) освоение теоретического материала;
- б) подготовка к практическим занятиям;
- в) работа с электронным учебно-методическим комплексом.

Самостоятельная работа студентов является составной частью учебной работы. Ее основной целью является формирование навыка самостоятельного приобретения знаний по некоторым не особо сложным вопросам теоретического курса, закрепление и углубление полученных знаний, самостоятельная работа со справочниками, периодическими изданиями и научно-популярной литературой, в том числе находящимися в глобальных компьютерных сетях.

9 Оценочные средства для текущего контроля успеваемости, промежуточной аттестации по итогам освоения дисциплины и учебно-методическое обеспечение самостоятельной работы студентов

Уровень и качество знаний студентов оцениваются по результатам текущего контроля успеваемости и промежуточной аттестации по итогам освоения дисциплины в виде экзамена.

Текущий контроль успеваемости обучающихся включает устные опросы и задания, выдаваемые на самостоятельную работу по темам дисциплины.

Устный опрос проводится на практическом занятии в течение не более 10 минут с целью контроля усвоения теоретического материала, излагаемого на лекции. Перечень вопросов определяется уровнем подготовки учебной группы, а также индивидуальными особенностями обучающихся.

Индивидуальное задание – это продукт самостоятельной работы обучающегося, представляющий собой самостоятельную работу по представлению полученных результатов решения определенного учебно-практического задания.

Контроль выполнения задания, выдаваемого на самостоятельную работу, преследует собой цель своевременного выявления плохо усвоенного материала дисциплины для последующей корректировки или организации обязательной консультации.

Итоговая аттестация по итогам освоения дисциплины проводится в форме экзамена.

Экзамен позволяет оценить уровень освоения компетенций за весь период изучения дисциплины. К моменту сдачи экзамена должны быть успешно пройдены предыдущие формы контроля. При проведении итоговой аттестации (экзамен) применяется тестирование.

9.1 Балльно-рейтинговая оценка текущего контроля успеваемости и знаний студентов по дисциплине

Не применяется.

9.2 Методические рекомендации по проведению процедуры оценивания знаний, умений и навыков и (или) опыта деятельности, характеризующих этапы формирования компетенций

Устный опрос оценивается следующим образом:

«зачтено»: обучающийся дает ответ на поставленный вопрос по существу и правильно отвечает на уточняющие вопросы;

«не зачтено»: обучающийся отказывается отвечать на поставленный вопрос, либо отвечает на него неверно и при формулировании дополнительных (вспомогательных) вопросов.

Решение задач оценивается:

«зачтено»: обучающийся самостоятельно правильно решает задачу, дает обоснованную оценку по итогу решения;

«не зачтено»: обучающийся отказывается от выполнения задачи или не способен ее решить самостоятельно, а также с помощью преподавателя.

9.3 Темы курсовых работ (проектов) по дисциплине

Написание курсовых работ (проектов) учебным планом не предусмотрено.

9.4 Контрольные вопросы для проведения входного контроля остаточных знаний по обеспечивающим дисциплинам

Входной контроль по дисциплине не предусмотрен.

9.5 Описание показателей и критериев оценивания компетенций на различных этапах их формирования, описание шкал оценивания

Компетенции	Показатели оценивания (индикаторы достижения) компетенций	Критерии оценивания
I этап		
ОПК-4	ID^1	Знает: - основные принципы самоорганизации и
	$ID^2_{ОПК\ 4}$	самообразования; - основные сведения о дискретных структурах, используемых в персональных компьютерах; - способы обработки материала с применением современных информационных технологий; - способы сбора библиографической информации по дисциплине; - структуру локальных и глобальных компьютерных сетей. Умеет: - воспринимать и реализовывать на практике полученные знания; - работать в качестве пользователя персонального компьютера;
II этап		
ОПК-4	$ID^1_{ОПК\ 4}$	Умеет: - обрабатывать и анализировать материал с применением современных информационных технологий;
	$ID^2_{ОПК\ 4}$	- работать с компьютером как средством управления информацией. Владеет: - методами сбора, хранения и обработки информации, применяемые в профессиональной деятельности; - методами самоорганизации и самообразования; - методами обработки материала, знаниями по информационной безопасности; - методами решения функциональных и

9.6 Типовые контрольные задания для проведения текущего контроля и промежуточной аттестации по итогам освоения дисциплины

Перечень типовых вопросов для проведения текущего контроля:

1. Что такое алгоритм? Какова его роль в автоматизации логистических процессов?
2. Для чего используются блок-схемы при визуализации алгоритмов?
3. Приведите примеры логистических задач, которые можно решить с помощью алгоритмов.
4. Какие существуют основные среды разработки для языка Python? В чём особенности Jupyter Notebook?
5. Из каких элементов состоит структура первой программы на Python?
6. Для чего используется функция `print()` в Python?
7. Что такое переменная? Как она хранится в памяти компьютера?
8. Перечислите основные типы данных в Python: строки и числа.
9. Как работает функция `input()`? Для чего она применяется?
10. Какие арифметические операторы используются в Python для выполнения расчётов?
11. Как построить простую формулу для расчёта стоимости перевозки на Python?
12. Что такое логика ветвления? Как работают операторы `if`, `else`, `elif`?
13. Какие операторы сравнения используются для проверки условий в Python?
14. Приведите пример программы, принимающей решения на основе входных данных.
15. Для чего используется цикл `for`? Как с его помощью перебирать последовательности (списки)?
16. Что такое функция `range()` и как она применяется для генерации числовых последовательностей?
17. Как с помощью цикла обработать список грузов или городов?
18. Что такое список (`list`) в Python? Каковы его особенности?
19. Для чего используются словари (`dict`) в Python? Как они устроены?
20. Что такое функция в программировании? Зачем создавать собственные функции?
21. Как осуществляется передача аргументов в функцию и возврат результата?
22. Что такое наука о данных? Какие задачи она решает?
23. Перечислите основные структуры данных: стек, массив, очередь, хэш-таблица.
24. Какие этапы включает работа с таблицами и подготовка данных для анализа?
25. Какие базовые статистические и логические функции используются в табличных процессорах?
26. В чём заключается роль визуализации в анализе логистических показателей?
27. Для чего используется библиотека `matplotlib` в Python?
28. Какие типы графиков можно построить с помощью `matplotlib`?
29. Для чего предназначена библиотека `Pandas` и какие основные задачи она решает при работе с данными?
30. Как осуществляется импорт библиотеки `Pandas` в Python и какие основные объекты она предоставляет для анализа данных?
31. Чем отличаются структуры данных `Series` и `DataFrame` в `Pandas`? Приведите примеры их использования.

32. Как создать объект DataFrame на основе списка, словаря или с помощью функции read_csv?
33. Какие методы в Pandas используются для получения информации о структуре и содержимом таблицы (info, describe)?
34. Как отличить числовые и категориальные признаки в данных с помощью Pandas?
35. Какие агрегирующие функции (value_counts, unique, nunique, groupby) применяются для анализа данных в Pandas?
36. Как использовать методы min(), max() и mean() для вычисления статистических показателей по столбцам DataFrame?
37. Как объединить две таблицы в Pandas с помощью метода merge? Какие параметры (on, how) при этом используются?
38. Какие библиотеки визуализации данных используются совместно с Pandas и как построить графики и диаграммы на основе DataFrame?

Темы докладов

Доклады не предусмотрены

Темы индивидуальных заданий

Задача1. Создайте список containers, в котором перечислите размеры 15–20 контейнеров (например, '20', '40'). Создайте пустой словарь count для подсчёта. С помощью цикла for пройдите по списку containers. Если размер '20', увеличьте счётчик для 20-футовых. Если размер '40', увеличьте счётчик для 40-футовых. Выведите результат: «20-футовых: [число], 40-футовых: [число]».

Задача2. Запросите у пользователя ввод с помощью функции input(). опрос: «Есть ли у вас паспорт и билет? (да/нет)». С помощью конструкции if-else проверьте ответ. Если ответ 'да', выведите: «Добро пожаловать на борт!». Если ответ 'нет', выведите: «Посадка запрещена.». Для любого другого ответа выведите: «Неверный ввод. Пожалуйста, ответьте 'да' или 'нет'.» (используйте elif).

Задача3. Создайте список cargo_list с названиями 10–12 разных грузов (например, 'зерно', 'уголь', 'автомобили'). Напишите функцию find_cargo(list, name). Внутри функции с помощью цикла for проверьте, есть ли название name в списке list. Если груз найден, верните из функции True. Если цикл закончился, а груз не нашли, верните False. В основной части программы вызовите эту функцию, чтобы найти груз 'зерно'. С помощью if выведите сообщение: «Груз найден» или «Груз не найден» в зависимости от того, что вернула функция.

Задача 4: Напишите программу, которая запрашивает у пользователя номер рейса и пункт назначения. Используя конструкцию with open('flights.log', 'a') as file, добавьте введённые данные в конец файла flights.log в формате: Номер рейса: [номер], Пункт назначения: [пункт]. Поместите код ввода данных в блок try. Если пользователь введёт пустую строку для номера рейса, сгенерируйте собственное исключение с помощью raise ValueError("Номер рейса не может быть пустым"). Используйте блок except ValueError as e, чтобы перехватить эту ошибку (и стандартную ошибку преобразования) и вывести понятное сообщение пользователю, не прерывая работу программы. Настройте модуль logging, чтобы

все ошибки (например, если файл нельзя открыть) записывались в отдельный файл `error.log`.

Задача 5. Работа с данными о грузоперевозках. Дан файл `cargo.csv` с данными о перевозках: город отправления, город назначения, вес груза, стоимость перевозки, дата отправки. Загрузите данные в `DataFrame` с помощью `Pandas`. Посмотрите первые 5 и последние 3 строки таблицы. Определите, сколько всего уникальных городов отправления и назначения. Найдите среднюю стоимость перевозки и максимальный вес груза. Отфильтруйте только те перевозки, где стоимость выше средней. Постройте гистограмму распределения веса грузов с помощью `Matplotlib`.

Задача 6. Агрегация и группировка данных. Дан файл `orders.csv` с информацией о заказах: ID заказа, город, категория товара, количество, сумма заказа. Загрузите данные в `DataFrame`. Посчитайте, сколько заказов было в каждом городе. Определите, какая категория товаров самая популярная (по количеству заказов). Для каждой категории вычислите среднюю сумму заказа. Объедините таблицу с файлом `cities.csv` (содержит город, регион) по полю «город» с помощью метода `merge`. Постройте столбчатую диаграмму: среднее количество заказов по регионам.

Задача 7. Очистка и преобразование данных. Дан файл `transport.csv` с данными о транспортных средствах: номер машины, тип транспорта, пробег (км), год выпуска, статус (в работе/на ремонте). Загрузите данные в `DataFrame`. Проверьте наличие пропусков и дубликатов. Удалите дубликаты и заполните пропуски в поле «год выпуска» средним значением. Преобразуйте значения в столбце «пробег» из строкового формата (например, «120 000 км») в числовой. Создайте новый столбец «возраст» — разница между текущим годом и годом выпуска. Отфильтруйте только транспорт типа «грузовик» и со статусом «в работе». Постройте диаграмму рассеяния: пробег vs возраст для грузовиков.

Вопросы для проведения промежуточной аттестации

Вопросы промежуточной аттестации идентичны перечню типовых вопросов для проведения текущего контроля.

10 Методические рекомендации для обучающихся по освоению дисциплины

Приступая к изучению дисциплины «Информатика», обучающемуся необходимо внимательно ознакомиться с тематическим планом занятий и списком рекомендованной литературы. Уровень и глубина усвоения дисциплины обучающегося зависят от активной и систематической работы на лекциях и практических занятиях. В этой связи важное значение имеет самостоятельная работа обучающегося. Целью этой работы является вовлечение обучающегося в самостоятельную познавательную деятельность и формирование у него методов организации своей деятельности, которые приводят к развитию самостоятельного мышления, способностей к профессиональному саморазвитию, самосовершенствованию и самореализации в современных условиях.

Основными видами аудиторной работы студентов являются лекции и практические занятия. В ходе лекции преподаватель излагает и разъясняет основные, наиболее сложные понятия, а также соответствующие теоретические и практические проблемы, дает задания и рекомендации для практических занятий, а также указания по выполнению обучающимся самостоятельной работы.

Задачами лекций являются:

- ознакомление обучающихся с целями, задачами и структурой дисциплины, ее местом в системе наук и связями с другими дисциплинами;
- краткое, но по существу, изложение комплекса основных понятий, подходов, методов, принципов данной дисциплины;
- краткое изложение наиболее существенных положений, раскрытие особенно сложных, актуальных вопросов.

Важное значение имеет формирование конспекта лекций. При его ведении необходимо четко фиксировать рубрикацию материала, т.е. разграничение разделов, тем, вопросов, параграфов и т. п. Необходимо делать специальные пометки, например, в случаях, когда какое-либо определение, положение, вывод остались неясными, сомнительными. Качественно сделанный конспект лекций поможет обучающемуся в процессе самостоятельной работы и при подготовке к сдаче экзамена.

Цели практических занятий: закрепить теоретические знания, полученные студентом на лекциях и в результате самостоятельного изучения соответствующих разделов рекомендуемой литературы. Темы практических занятий заранее сообщаются обучающимся для того, чтобы они имели возможность подготовиться и проработать соответствующие теоретические вопросы дисциплины. В начале каждого практического занятия преподаватель:

- кратко доводит до обучающихся цели и задачи занятия, обращая их внимание на наиболее сложные вопросы по изучаемой теме;
- проводит устный опрос обучающихся, в ходе которого также обсуждаются наиболее сложные вопросы.

Самостоятельная работа обучающегося весьма многообразна и содержательна. Она включает следующие виды занятий (п. 5.6):

- самостоятельный поиск, анализ информации и проработка учебного материала;
- подготовку к устному и ИЗ.

Завершающим этапом самостоятельной работы является подготовка к сдаче экзамена по дисциплине, предполагающая интеграцию и систематизацию всех полученных при изучении учебной дисциплины знаний. К моменту сдачи экзамена должны быть успешно пройдены предыдущие формы контроля.

Экзамен позволяет определить уровень освоения обучающимся компетенций за периоды изучения данной дисциплины. Промежуточная аттестация проводится в виде теста с решением типовой задачи.

Рабочая программа дисциплины составлена в соответствии с требованиями ФГОС ВО по направлению подготовки 23.03.01 «Технология транспортных процессов».

Программа рассмотрена и утверждена на заседании кафедры № 8 «Прикладной математики и информатики» «31» марта 2026 года, протокол №11.

Разработчики:

ст. преп.

(ученая степень, ученое звание, фамилия и инициалы разработчиков)

Родионова Ю.И.

И.о. заведующего кафедрой № 8

к.т.н.

(ученая степень, ученое звание, фамилия и инициалы разработчиков)

Земсков Ю.В.

Программа согласована:

Руководитель ОПОП

к.н., доцент

(ученая степень, ученое звание, фамилия и инициалы руководителя ОПОП)

Иванова Н.В.

Программа рассмотрена и согласована на заседании Учебно-методического совета Университета «22» апреля 2026 года, протокол № 7.