

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ
ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ АВТОНОМНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ
«Национальный исследовательский ядерный университет «МИФИ»
Обнинский институт атомной энергетики –
филиал федерального государственного автономного образовательного учреждения высшего образования
«Национальный исследовательский ядерный университет «МИФИ»
(ИАТЭ НИЯУ МИФИ)

Утверждено на заседании
УМС ИАТЭ НИЯУ МИФИ
Протокол №2-8/2024 От 30.08.2024

РАБОЧАЯ ПРОГРАММА УЧЕБНОЙ ДИСЦИПЛИНЫ

Python для анализа данных

Шифр, название дисциплины

01.04.02 «Прикладная математика и информатика»

Шифр, название специальности/направления подготовки

Математическое моделирование и прикладной анализ данных

Название программы магистратуры

магистр

(Квалификация (степень) выпускника)

Форма обучения: очная

г. Обнинск 2024 г.

Программа составлена в соответствии с требованиями образовательного стандарта высшего образования национального исследовательского ядерного университета «МИФИ» по направлению подготовки 01.04.02 – Прикладная математика и информатика. (квалификация (степень) магистр).

Программу составил:

_____ С.В. Ермаков, доцент, к.ф.-м.н, доцент

Рецензент:

_____ Г.Е. Деев, доцент, к.ф.-м.н, доцент

Программа рассмотрена на заседании ОИКС

(протокол № 5/7 от «30» июля от 2024 г.)

Руководитель направления подготовки 01.04.02
«Прикладная математика и информатика»

_____ Ермаков С.В.

« ____ » _____ 2024 г.

1. Перечень планируемых результатов обучения по дисциплине, соотнесенных с планируемыми результатами освоения образовательной программы

В результате освоения ООП магистратуры обучающийся должен овладеть следующими результатами обучения по дисциплине:

Коды компетенций	Результаты освоения ООП Содержание компетенций*	Перечень планируемых результатов обучения по дисциплине**
УК-1	Способен осуществлять критический анализ проблемных ситуаций на основе системного подхода, выработать стратегию действий	З-УК-1 Знать: методы системного и критического анализа; методики разработки стратегии действий для выявления и решения проблемной ситуации У-УК-1 Уметь: применять методы системного подхода и критического анализа проблемных ситуаций; разрабатывать стратегию действий, принимать конкретные решения для ее реализации В-УК-1 Владеть: методологией системного и критического анализа проблемных ситуаций; методиками постановки цели, определения способов ее достижения, разработки стратегий действий.
УК-3	Способен организовывать и руководить работой команды, выработывая командную стратегию для достижения поставленной цели	З-УК-3 Знать: методики формирования команд; методы эффективного руководства коллективами; основные теории лидерства и стили руководства У-УК-3 Уметь: разрабатывать план групповых и организационных коммуникаций при подготовке и выполнении проекта; сформулировать задачи членам команды для достижения поставленной цели; разрабатывать командную стратегию; применять эффективные стили руководства командой для достижения поставленной цели В-УК-3 Владеть: умением анализировать, проектировать и организовывать межличностные, групповые и организационные коммуникации в команде для достижения поставленной цели; методами организации и управления коллективом

2. Место дисциплины в структуре ООП магистратуры

Дисциплина реализуется в рамках общенаучного модуля.

Для освоения дисциплины необходимы компетенции, сформированные в рамках изучения следующих дисциплин: Базовый Python + объектно-ориентированное программирование (ООП)

в Python, Математика для анализа данных

Дисциплина изучается на 1 курсе в 2 семестре.

3. Объем дисциплины в зачетных единицах с указанием количества академических часов, выделенных на контактную работу обучающихся с преподавателем (по видам занятий) и на самостоятельную работу обучающихся

Общая трудоемкость (объем) дисциплины составляет 3 зачетных единиц (з.е.), 108 академических часа.

3.1. Объем дисциплины по видам учебных занятий (в часах)

	Семестр		
	№ 1	№ 2	Всего
	Количество часов на вид работы:		
Контактная работа обучающихся с преподавателем			
Аудиторные занятия (всего)		64	64
В том числе:			
<i>лекции</i>		32	32
<i>практические занятия</i>		32	32
<i>лабораторные занятия</i>			
Промежуточная аттестация			
В том числе:			
<i>зачет</i>			
<i>экзамен</i>		36	36
Самостоятельная работа обучающихся (всего)		8	8
В том числе:			
<i>проработка учебного (теоретического) материала</i>		2	2
<i>выполнение индивидуальных заданий</i>		2	2
<i>подготовка ко всем видам контрольных испытаний текущего контроля успеваемости (в течение семестра)</i>		2	2
<i>подготовка ко всем видам контрольных испытаний промежуточной аттестации (по окончании семестра)</i>		2	2
<i>Всего (часы):</i>		108	108
<i>Всего (зачетные единицы):</i>		3	3

4. Содержание дисциплины, структурированное по темам (разделам) с указанием отведенного на них количества академических часов и видов учебных занятий

4.1. Разделы дисциплины и трудоемкость по видам учебных занятий (в академических часах)

№ п/п	Наименование раздела /темы дисциплины	Общая трудоёмкость всего (в часах)	Виды учебных занятий, включая самостоятельную работу обучающихся и трудоемкость (в часах)				Формы текущего контроля успеваемости
			Аудиторные учебные занятия			СРО	
			Лек	Сем/Пр	Лаб		
1.			32	32	-	8	
1	Подмодуль «Numpy»						
1.1.	Введение	4	1	1	-		
1.2.	Основные операции с массивами	6	2	2	-		
1.3.	Основные функции	4	1	1	-	2	
1.4.	Статистика	4	1	1			
1.5.	Загрузка и сохранение данных	4	1	1	-		
1.6.	Линейная алгебра	4	1	1	-	2	
1.7.	Логические функции	4	1	1	-		
1.8.	Случайные величины	4	1	1		2	
1.9.	Практический кейс	4	1	1			
2	Подмодуль «Pandas»				-		
2.1	Введение в Pandas	4	1	1			
2.2.	Загружаем и обрабатываем данные	4	1	1			
2.3.	Изучаем датасет - проводим исследовательский анализ данных (EDA)	3	1	1			
2.4.	Расчеты метрик	3	1	1		2	
2.5.	Дополнительные методы	3	1	1			
2.6	Практический блиц-тест	3	1	1			
2.7	Практический кейс	3	1	1			
2.8	Несколько лайфхаков от Андрона	3	1	1			
3	Подмодуль «Визуализация в Python»						
3.1	Matplotlib: Основы	3	1	1			
3.2	Matplotlib: Тюнинг графиков	3	1	1			
3.3	Matplotlib: Остальные виды визуализаций	3	1	1			
3.4	Matplotlib: Совмещение графиков	3	1	1			
3.5	Seaborn: Основы	3	1	1			
3.6	Seaborn: Продвинутая визуализация	3	1	1			

3.7	Plotly: Интерактивные графики	3	1	1			
3.8	Практическая работа	3	1	1			
3.9	Разбор практической работы	3	1	1			
3.10	Полезные советы	3	1	1			
3.11	Итоговый проект	5	1	1			
3.12	Итоговый проект - Тестовое в Самокат	4	1	1			
3.13	Итоговый проект - Разбор	5	2	2			

4.2. Содержание дисциплины, структурированное по разделам (темам)

Лекционный курс

№	Наименование раздела /темы дисциплины	Содержание
1	Подмодуль «NumPy»	
1.1.	Введение	Введение в NumPy
1.2.	Основные операции с массивами	основами использования библиотеки NumPy
1.3.	Основные функции	Основные функции, Broadcasting - специальной техники, позволяющей выполнять операции над массивами разных форм
1.4.	Статистика	статистика и анализ, с помощью инструментов библиотеки NumPy. Функции, которые позволяют нам определить средние значения данных. Меры, которые помогут нам оценить, насколько данные распределены вокруг среднего значения. Узнаем, как вычислять корреляцию между наборами данных.
1.5.	Загрузка и сохранение данных	Загрузка и сохранение данных
1.6.	Линейная алгебра	обзор ключевых понятий линейной алгебры, таких как векторы, матрицы, операции над ними и базовые преобразования. Изучим разнообразные матричные операции, такие как умножение, транспонирование, обратные матрицы и детерминанты. Научимся использовать методы NumPy для решения систем линейных уравнений
1.7.	Логические функции	мощные инструменты, которые NumPy предоставляет для работы с логическими данными. Как выполнять операции с булевыми значениями, применять логические функции к массивам
1.8.	Случайные величины	Случайности и вероятностные распределения. Как создавать случайные данные, моделировать разнообразные вероятностные события и анализировать их с помощью NumPy.
1.9.	Практический кейс	практический кейс

2	Подмодуль «Pandas»	
2.1	Введение в Pandas	Введение в Pandas
2.2.	Загружаем и обрабатываем данные	read_csv, кодировка head, tail set_option, get_option, reset_option Меняем названия, порядок и количество столбцов Полезные атрибуты (shape, columns, index, dtypes, size) Меняем типы данных Меняем формат даты Немного про работу с датой в Pandas replace unique values Обработка пропусков - ищем их, fillna, interpolate, dropna
2.3.	Изучаем датасет - проводим исследовательский анализ данных (EDA)	проводим исследовательский анализ данных (EDA)
2.4.	Расчеты метрик	посчитаем: Классический ABC-анализ Многомерный ABC-анализ Создадим функцию для универсального расчета ABC-анализа Динамику продаж XYZ-анализ
2.5.	Дополнительные методы	методы и функции, которые не состоят в какой-то группе и стоят обособленно.
2.6	Практический блиц-тест	тест
2.7	Практический кейс	практический кейс
2.8	Несколько лайфхаков от Андрона	Несколько лайфхаков от Андрона
3	Подмодуль «Визуализация в Python»	
3.1	Matplotlib: Основы	Коротко про библиотеку Загружаем данные Строим линейный график - динамика попыток по дате Строим другой линейный график - зависимость попыток от месяца
3.2	Matplotlib: Тюнинг графиков	Построение scatterplot Прозрачность Маркеры Цвета Название графика и осей Отсечки по осям X и Y Как убрать оси? Сетка Добавление средней линии Аннотации
3.3	Matplotlib: Остальные виды визуализаций	Строим график динамики попыток по языкам, добавляем легенду

		Попытки по месяцам - барчарт + линейный график Горизонтальный барчарт Гистограмма по задачам Боксплот попыток по дням Пайчарт по языку
3.4	Matplotlib: Совмещение графиков	subplots В чем разница между plt и ax Создание нескольких графиков tight_layout twinx и get_legend_handles_labels rcParams сохранение картинки
3.5	Seaborn: Основы	lineplot scatterplot подписи к осям цвета прозрачность hue
3.6	Seaborn: Продвинутая визуализация	countplot heatmap на примере корреляций pairplot jointplot boxplot violinplot warmplot
3.7	Plotly: Интерактивные графики	библиотека для построения визуализаций - Plotly
3.8	Практическая работа	Практическая работа
3.9	Разбор практической работы	Разбор практической работы
3.10	Полезные советы	Полезные советы
3.11	Итоговый проект	Итоговый проект
3.12	Итоговый проект - Тестовое в Самокат	Итоговый проект - Тестовое в Самокат
3.13	Итоговый проект - Разбор	Итоговый проект - Разбор

Практические/семинарские занятия

№	Наименование раздела /темы дисциплины	Содержание
1	Подмодуль «NumPy»	
1.1.	Введение	Введение в NumPy
1.2.	Основные операции с массивами	основами использования библиотеки NumPy
1.3.	Основные функции	Основные функции, Broadcasting - специальной техники,

		позволяющей выполнять операции над массивами разных форм
1.4.	Статистика	статистика и анализ, с помощью инструментов библиотеки NumPy. Функции, которые позволяют нам определить средние значения данных. Меры, которые помогут нам оценить, насколько данные распределены вокруг среднего значения. Узнаем, как вычислять корреляцию между наборами данных.
1.5.	Загрузка и сохранение данных	Загрузка и сохранение данных
1.6.	Линейная алгебра	обзор ключевых понятий линейной алгебры, таких как векторы, матрицы, операции над ними и базовые преобразования. Изучим разнообразные матричные операции, такие как умножение, транспонирование, обратные матрицы и детерминанты. Научимся использовать методы NumPy для решения систем линейных уравнений
1.7.	Логические функции	мощные инструменты, которые NumPy предоставляет для работы с логическими данными. Как выполнять операции с булевыми значениями, применять логические функции к массивам
1.8.	Случайные величины	Случайности и вероятностные распределения. Как создавать случайные данные, моделировать разнообразные вероятностные события и анализировать их с помощью NumPy.
1.9.	Практический кейс	практический кейс
2	Подмодуль «Pandas»	
2.1	Введение в Pandas	Введение в Pandas
2.2.	Загружаем и обрабатываем данные	read_csv, кодировка head, tail set_option, get_option, reset_option Меняем названия, порядок и количество столбцов Полезные атрибуты (shape, columns, index, dtypes, size) Меняем типы данных Меняем формат даты Немного про работу с датой в Pandas replace unique values Обработка пропусков - ищем их, fillna, interpolate, dropna
2.3.	Изучаем датасет - проводим исследовательский анализ данных (EDA)	проводим исследовательский анализ данных (EDA)
2.4.	Расчеты метрик	посчитаем: Классический ABC-анализ Многомерный ABC-анализ Создадим функцию для универсального расчета ABC-анализа Динамику продаж XYZ-анализ

2.5.	Дополнительные методы	методы и функции, которые не состоят в какой-то группе и стоят обособленно.
2.6	Практический блиц-тест	тест
2.7	Практический кейс	практический кейс
2.8	Несколько лайфхаков от Андрона	Несколько лайфхаков от Андрона
3	Подмодуль «Визуализация в Python»	
3.1	Matplotlib: Основы	Коротко про библиотеку Загружаем данные Строим линейный график - динамика попыток по дате Строим другой линейный график - зависимость попыток от месяца
3.2	Matplotlib: Тюнинг графиков	Построение scatterplot Прозрачность Маркеры Цвета Название графика и осей Отсечки по осям X и Y Как убрать оси? Сетка Добавление средней линии Аннотации
3.3	Matplotlib: Остальные виды визуализаций	Строим график динамики попыток по языкам, добавляем легенду Попытки по месяцам - барчарт + линейный график Горизонтальный барчарт Гистограмма по задачам Боксплот попыток по дням Пайчарт по языку
3.4	Matplotlib: Совмещение графиков	subplots В чем разница между plt и ax Создание нескольких графиков tight_layout twinx и get_legend_handles_labels rcParams сохранение картинки
3.5	Seaborn: Основы	lineplot scatterplot подписи к осям цвета прозрачность hue
3.6	Seaborn: Продвинутая визуализация	countplot heatmap на примере корреляций pairplot jointplot boxplot violinplot

		warmplot
3.7	Plotly: Интерактивные графики	библиотека для построения визуализаций - Plotly
3.8	Практическая работа	Практическая работа
3.9	Разбор практической работы	Разбор практической работы
3.10	Полезные советы	Полезные советы
3.11	Итоговый проект	Итоговый проект
3.12	Итоговый проект - Тестовое в Самокат	Итоговый проект - Тестовое в Самокат
3.13	Итоговый проект - Разбор	Итоговый проект - Разбор

Лабораторные занятия

Не предусмотрены.

5. Перечень учебно-методического обеспечения для самостоятельной работы обучающихся по дисциплине

В качестве учебно-методических материалов используется рекомендованная литература.

6. Фонд оценочных средств для проведения промежуточной аттестации обучающихся по дисциплине

6.1. Паспорт фонда оценочных средств по дисциплине

№ п/п	Контролируемые разделы (темы) дисциплины (результаты по разделам)	Код контролируемой компетенции (или её части) / и ее формулировка	Наименование оценочного средства
1.5	Загрузка и сохранение данных	УК-3	Контрольная работа № 1
1.8.	Случайные величины	УК-3	Контрольная работа № 1
2.3.	Свертка обобщенных функций	УК-1	Контрольная работа № 2
2.4.	Расчеты метрик	УК-1	Контрольная работа № 2
3.7.	Plotly: Интерактивные графики	УК-3	Контрольная работа № 2

6.2. Типовые контрольные задания или иные материалы

6.2.1. Экзамен

В экзаменационном билете два теоретических вопроса и один практический

Теоретические вопросы билета:

Теоретические вопросы билета:

1. Какие основные операции можно выполнять с массивами NumPy?
2. Как создать и использовать функции в NumPy для выполнения математических операций (например, суммирование, нахождение среднего)?

3. Какие статистические функции предоставляет NumPy, и как их применить для анализа данных?
4. Как в NumPy загрузить данные из файла и сохранить массив в файл?
5. Как выполнить линейные алгебраические операции (например, умножение матриц) с помощью NumPy?
6. Как использовать логические операции и маски для фильтрации данных в NumPy?
7. Как сгенерировать случайные величины с использованием NumPy? Приведите пример.
8. Что такое DataFrame в Pandas, и чем он отличается от NumPy массива?
9. Как загрузить данные в Pandas из различных источников (например, CSV, Excel, SQL)?
10. Какие методы Pandas можно использовать для предварительной обработки данных (например, очистка данных, заполнение пропусков)?
11. Как выполнять исследовательский анализ данных (EDA) с использованием Pandas? Какие методы для этого могут быть полезны?
12. Как рассчитывать агрегированные метрики, такие как сумма, среднее, медиана, с использованием Pandas?
13. Какие дополнительные методы Pandas можно использовать для манипуляции данными (например, группировка, сортировка, фильтрация)?
14. Какие типы графиков можно построить с помощью Matplotlib и какие параметры можно настроить для их кастомизации?
15. Как улучшить визуализацию в Matplotlib, настроив внешний вид графиков (например, цвета, шрифты, легенды)?
16. Какие типы графиков можно построить с помощью Matplotlib, помимо стандартных линейных графиков (например, гистограммы, диаграммы рассеяния)?
17. Как совмещать несколько графиков на одном изображении в Matplotlib и какие функции для этого используются?
18. Как использовать Seaborn для создания более сложных визуализаций, таких как тепловые карты и парные диаграммы?
19. В чем преимущества Seaborn по сравнению с Matplotlib для визуализации статистических данных?
20. Какие возможности для создания интерактивных графиков предоставляет Plotly?
21. Как добавить элементы интерактивности в графики Plotly (например, всплывающие подсказки, изменения масштаба)?
22. Как использовать Plotly для создания динамических и визуально привлекательных дашбордов?
23. Как с помощью Matplotlib, Seaborn и Plotly можно визуализировать временные ряды и распределения данных?
24. Как выбрать подходящий тип визуализации данных в зависимости от поставленной задачи (например, для сравнения категорий, временных рядов и т. д.)?

Критерий оценки – правильность и полнота ответа на вопросы. Оценка выставляется по шкале от 0 до 40 баллов: теоретические вопросы –30 баллов, 10 баллов– дополнительные вопросы. Экзамен считается сданным при оценке не ниже 25 баллов.

6.2.2. Контрольная работа № 1

В вашей рабочей папке лежит файл itresume-users-pandas.csv. В нем содержится

зашифрованная информация о некоторых пользователях платформы **IT Resume**:

- идентификатор пользователя в базе данных
- зашифрованный логин пользователя
- дата регистрации на платформе

Данные уже загружены в рабочую область - вы можете их считывать с помощью Python. Например, вы можете ввести в терминале справа функцию чтения этого файла и она у вас отработает корректно.

6.2.2. Контрольная работа № 2

В вашей рабочей папке лежит файл itresume-coderun.csv. В нем содержится кусочек информации о том, как пользователи платформы **IT Resume** отправляют код на выполнение (нажимают на кнопку Проверить):

- идентификатор записи
- дата и время отправки кода на выполнение
- идентификатор задачи
- идентификатор пользователя
- идентификатор языка

Данные уже загружены в рабочую область - вы можете их считывать с помощью Python. Например, вы можете ввести в терминале справа функцию чтения этого файла и она у вас отработает корректно.

б) критерии оценивания компетенций (результатов) – правильная работа кода программы, понимание алгоритма метода оптимизации, умение вывести необходимые для алгоритма формулы.

в) описание шкалы оценивания:

Каждая задача оценивается по шкале от 0 до 10 баллов.

Контрольная работа считается выполненной успешно при суммарной оценке не ниже 18 баллов.

6.3. Методические материалы, определяющие процедуры оценивания знаний, умений, навыков и (или) опыта деятельности, характеризующих этапы формирования компетенций

Форма аттестации	Наименование оценочного средства	Баллы
Экзамен (100 баллов)	Контрольная работа № 1	30
	Контрольная работа № 2	30
	Ответы на экзаменационный билет	40

7. Перечень основной и дополнительной учебной литературы, необходимой для освоения дисциплины

а) основная учебная литература:

1. Хейдт, Груздев: Изучаем pandas. 2019
2. Python для сложных задач: наука о данных. 2-е межд. изд. Вандер Плас Д. 2024
3. Python и анализ данных Уэс Маккинни. 2022

4. Python for Data Analysis: Data Wrangling with Pandas, NumPy, and IPython 2nd Edition by Wes McKinney. 2017

б) дополнительная учебная литература:

1. Юрий Орлов — Python: от новичка до эксперта — 2021 — 350 стр.
2. Иван Журавлев — Мастерство Python: от основ до продвинутых техник — 2022 — 400 стр.
3. Екатерина Химченко — Основы ООП и Python: от классов до декораторов — 2023 — 280 стр.
4. Александр Черняев — Многозадачность и асинхронность в Python — 2020 — 220 стр.

8. Перечень ресурсов* информационно-телекоммуникационной сети «Интернет» (далее - сеть «Интернет»), необходимых для освоения дисциплины

-

9. Методические указания для обучающихся по освоению дисциплины

Вид учебного занятия	Организация деятельности студента
Лекция	Написание конспекта лекций: кратко, схематично, последовательно фиксировать основные положения, выводы, формулировки, обобщения; пометить важные мысли, выделять ключевые слова, термины. Проверка терминов, понятий с помощью энциклопедий, словарей, справочников с выписыванием толкований в тетрадь. Обозначить вопросы, термины, материал, который вызывает трудности, пометить и попытаться найти ответ в рекомендуемой литературе. Если самостоятельно не удастся разобраться в материале, необходимо сформулировать вопрос и задать преподавателю на консультации, на практическом занятии.
Практические занятия	Проработка рабочей программы, уделяя особое внимание целям и задачам, структуре и содержанию дисциплины. Работа с конспектом лекций, просмотр рекомендуемой литературы. Изучение выбранной предметной области на примерах решения задач семинарских занятий, индивидуальных домашних заданий.
Курсовая работа	Не предусмотрена
Контрольная работа	Ознакомиться с основной и дополнительной литературой, включая справочные издания, зарубежные источники, основополагающие термины. Попрактиковаться в решении аналогичных домашних задач по всем темам контрольных работ.
Лабораторная работа	Не предусмотрена.
Подготовка к экзамену	При подготовке к экзамену необходимо ориентироваться на конспекты лекций и рекомендуемую литературу.

10. Перечень информационных технологий, используемых при осуществлении образовательного процесса по дисциплине, включая перечень программного обеспечения и информационных справочных систем (при необходимости)

Издательская система LaTeX для подготовки докладов, презентаций и учебного материала.

11. Описание материально-технической базы, необходимой для осуществления образовательного процесса по дисциплине

Видеопроектор, компьютер, издательская система LaTeX для подготовки докладов, презентаций и учебного материала.

12. Иные сведения и (или) материалы

12.1. Перечень образовательных технологий, используемых при осуществлении образовательного процесса по дисциплине

Часов в интерактивной форме – 8.

В ходе практических занятий происходит публичное обсуждение каждой решаемой задачи. При этом студенты высказывают свои мнения по выбору наиболее простого способа поиска оптимального решения.

После решения домашних работ на консультациях проводится разбор допущенных студентами ошибок.

12.2. Формы организации самостоятельной работы обучающихся (темы, выносимые для самостоятельного изучения; вопросы для самоконтроля; типовые задания для самопроверки)

Некоторые темы изучаются студентами самостоятельно. Для изучения используется приведённая в списке основная и дополнительная литература. Контроль освоения материала осуществляется при проверке контрольных работ, домашнего задания и на экзамене.

№	Тема и часть, изучаемая (осваиваемая) самостоятельно
1.1	Временные ряды и анализ временных данных
1.2	SQLAlchemy и работа с SQL в pandas
1.3	Работа с большими данными в Python (Dask, Vaex)
1.4	Визуализация данных с помощью Vokeh

Вопросы и задания для самоконтроля по всем темам:

1. Что такое массив в Numpy и чем он отличается от стандартных списков Python?
2. Как создать одномерный и двумерный массив в Numpy?
3. Какие основные операции можно выполнять с массивами Numpy?
4. Какие функции в Numpy используются для выполнения математических вычислений (например, суммирование, среднее)?
5. Как использовать Numpy для вычисления статистических показателей (например, медианы, стандартного отклонения)?
6. Как загружать данные в массивы Numpy и сохранять их в файл?
7. Что такое линейная алгебра в контексте Numpy, и какие функции для этого существуют?
8. Как работать с логическими операциями и масками в Numpy?
9. Как сгенерировать случайные величины с использованием Numpy?
10. Приведите пример практического использования Numpy в обработке данных.
11. Как создать DataFrame в Pandas и чем он отличается от Numpy массива?
12. Как загружать и обрабатывать данные с помощью Pandas?
13. Что такое исследовательский анализ данных (EDA) и какие методы можно использовать в Pandas для его выполнения?
14. Как рассчитывать метрики и агрегировать данные в Pandas (например, среднее, сумма, медиана)?

15. Какие дополнительные методы для работы с данными предоставляет Pandas (например, сортировка, фильтрация)?
16. В чем заключается задача практического блиц-теста с использованием Pandas?
17. Как можно эффективно использовать Pandas для анализа больших наборов данных?
18. Какие полезные советы или лайфхаки можно применить при работе с Pandas?
19. Какие основные функции предоставляет Matplotlib для создания графиков?
20. Как настроить внешний вид графиков в Matplotlib (например, цвет, стиль линий, шрифты)?
21. Какие типы графиков можно строить с использованием Matplotlib (например, линейный график, гистограмма, диаграмма рассеяния)?
22. Как совмещать несколько графиков на одном изображении в Matplotlib?
23. Какие особенности работы с Seaborn для визуализации данных?
24. Как использовать Seaborn для построения более сложных и информативных графиков?
25. Как создать интерактивные графики с помощью Plotly?
26. Какие дополнительные инструменты и функции предоставляет Plotly для настройки графиков?
27. Что такое автоматизация в контексте обработки данных и как Python может помочь в этом?
28. Какие методы деплоя существуют для Python-программ, и как они применяются для моделей машинного обучения?

12.3. Краткий терминологический словарь

EDA	Исследовательский анализ данных — это процесс первичной обработки и анализа данных с целью выявления их структуры, закономерностей, аномалий и взаимосвязей. EDA включает в себя визуализацию данных, описание статистических характеристик, такие как среднее, медиана, стандартное отклонение, а также проверку гипотез и выявление возможных ошибок или недочетов в данных.
Plotly	Интерактивные графики — это графики, которые позволяют пользователю взаимодействовать с данными через различные элементы управления, такие как масштабирование, панорамирование, отображение всплывающих подсказок и фильтрацию данных. Plotly — это библиотека для создания таких графиков в Python, поддерживающая не только статичные визуализации, но и динамичные, с возможностью интерактивного взаимодействия, что особенно полезно при анализе больших и сложных наборов данных.