

Министерство науки и высшего образования Российской Федерации
НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ
ТОМСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ (НИ ТГУ)

Центр сопровождения образовательных инициативных проектов

УТВЕРЖДЕНО:
Руководитель сетевой ОПОП
З.И. Резанова

Рабочая программа дисциплины

Программирование на Python: базовый курс

по направлению подготовки

45.04.03 Фундаментальная и прикладная лингвистика

Направленность (профиль) подготовки :
Анализ естественного языка (NLP) в лингвистике и IT

Форма обучения
Очная

Квалификация
Магистр

Год приема
2023

1. Цель и планируемые результаты освоения дисциплины

Целью освоения дисциплины является формирование следующих компетенций:

ОПК-6. Способен осуществлять эффективное управление разработкой программных средств и информационных проектов в сфере своей профессиональной деятельности.

ПК-1. Способен исследовать и разрабатывать архитектуры систем искусственного интеллекта для задач анализа естественного языка на основе комплексов методов инструментальных средств систем искусственного интеллекта.

ПК-3. Способен разрабатывать и применять методы и алгоритмы машинного обучения для решения задач в области обработки естественного языка.

ПК-4. Способен руководить проектами по созданию комплексных систем искусственного интеллекта.

УК-1. Способен осуществлять критический анализ проблемных ситуаций на основе системного подхода, вырабатывать стратегию действий.

УК-6. Способен определять и реализовывать приоритеты собственной деятельности и способы ее совершенствования на основе самооценки.

Результатами освоения дисциплины являются следующие индикаторы достижения компетенций:

ИОПК-6.1. Аргументированно выбирает математические и лингвистические методы решения профессиональных задач с применением языков программирования

ИПК-1.1. Исследует и анализирует контекст, запросы и требования к архитектурам систем искусственного интеллекта для задач анализа естественного языка

ИПК-1.2. Разрабатывает архитектуры систем искусственного интеллекта на основе комплексов методов инструментальных средств систем искусственного интеллекта, учитывая результаты исследования

ИПК-3.1. Планирует разработку или совершенствование методов и алгоритмов для решения комплекса задач предметной области на основе исследований, анализа, запросов. Выбирает и обосновывает выбранный вариант методов и алгоритмов

ИПК-3.2. Применяет методы и алгоритмы машинного обучения для решения задач в области обработки естественного языка

ИПК-4.2. Руководит процессом разработки архитектуры комплексных систем искусственного интеллекта, контролирует и анализирует результаты работы по проектам

ИУК-1.1. Выявляет проблемную ситуацию, на основе системного подхода осуществляет её многофакторный анализ и диагностику

ИУК-1.2. Осуществляет поиск, отбор и систематизацию информации для определения альтернативных вариантов стратегических решений в проблемной ситуации

ИУК-1.3. Предлагает и обосновывает стратегию действий с учетом ограничений, рисков и возможных последствий

ИУК-6.1. Разрабатывает стратегию личного и профессионального развития на основе соотнесения собственных целей и возможностей с развитием избранной сферы профессиональной деятельности.

2. Задачи освоения дисциплины

- освоить навыки программирования на языке Python;
- приобрести знания о базовых концепциях программирования на Python, областях его применимости, конструкциях языка Python;
- научиться проектировать архитектуру и реализовывать алгоритмические и программные решения системного и прикладного программного обеспечения
- развить и научиться использовать математических и информационных инструментальных средств, автоматизированных систем в научной и практической деятельности.

3. Место дисциплины в структуре образовательной программы

Дисциплина относится к Блоку 1 «Дисциплина (модули)».

Дисциплина относится к обязательной части образовательной программы.

4. Семестр(ы) освоения и форма(ы) промежуточной аттестации по дисциплине

Семестр 1, экзамен

5. Входные требования для освоения дисциплины

Для успешного освоения дисциплины требуются компетенции, сформированные в ходе освоения образовательных программ предшествующего уровня образования.

6. Язык реализации

Русский

7. Объем дисциплины

Общая трудоемкость дисциплины составляет 5 з.е., 180 часов, из которых:

-лекции: 30 ч.

-практические занятия: 24 ч.

в том числе практическая подготовка: 0 ч.

Объем самостоятельной работы студента определен учебным планом.

8. Содержание дисциплины, структурированное по темам

Модуль 1. Основы Python.

Тема 1. Введение

Краткое описание целей и задач, которые будут решаться на протяжении всего модуля.

Тема 2. Введение в программирование на Python

Python среди других языков программирования. Понятие “язык программирования”. Понятие “высокоуровневый язык”. Понятие “интерпретируемый язык”. Понятие “кроссплатформенный язык”. Синтаксис языка программирования. Плюсы и минусы Python. Версии Python.

Тема 3. Среда разработки: IT Resume

Среда разработки. IDE и CDE. Что такое редактор кода от It Resume и зачем его использовать? Задания с автопроверкой кода.

Тема 4. Среда разработки: Google Colab

Что такое GOOGLE COLAB и зачем его использовать? преимущества GOOGLE COLAB. Основы работы в GOOGLE COLAB.

Тема 5. Переменные и ввод/вывод данных

Оператор присваивания. Ввод данных пользователем и вывод на экран: функции input() и print().

Тема 6. Типы данных и их классификация

Как определить тип данных переменной? Классификация типов данных.

Тема 7. Числовые типы данных

Целые числа. Числа с плавающей точкой. Округление чисел. Логические переменные.

Тема 8. Типы данных: строки

Извлечение подстрок: срезы. Операции и функции для работы со строками. Методы для работы со строками. Форматирование строк. Преобразование строк в числа и наоборот.

Тема 9. Как работать с документацией Python

Раздел about. Раздел documentation.

Модуль 2.1 Погружение в типы данных.

Тема 1. Списки и кортежи

Индексация. списки: `.append()`, `.clear()`, `.count()`, `.copy()`, `.extend()`, `.reverse()`, `.sort()`.
Строки — это списки. Кортеж.

Тема 2. Типы данных: словарь

Методы работы со словарями: `.clear()`, `.keys()`, `.values()`, `.get()`, `.update()`, `.pop()`, `.setdefault()`.

Тема 3. Типы данных: множества

Зачем нужны множества, если есть списки? Задание множества. Методы множеств.

Тема 4. Приведение типов

Преобразование чисел между собой. Преобразование чисел в строки. Преобразование строк в числа. Преобразование в кортежи и списки.

Модуль 2.2 Условные операторы

Тема 1. Введение

Краткое описание целей и задач, которые будут решаться на протяжении всего модуля.

Тема 2. Операторы сравнения и логические операторы

Операторы сравнения. Логические операторы. Оператор тождественности `is`.

Тема 3. Практические примеры с различными операторами

Проверка на входжение элемента в последовательность. Проверка делимости чисел. Проверка эквивалентности.

Тема 4. Условный оператор

Простой условный оператор. Примеры использования условного оператора. Вложенные условные операторы. Неявное приведение к булеву типу.

Тема 5. Практические примеры с условным оператором

Использование логических операторов и операторов сравнения в условном операторе. Конструкция `if-elif-else`. Комплексные задачи. Практическое задание: подготовка данных с резюме.

Тема 6. Исключения

Понятие исключения. Обработка исключений. Генерация исключений.

Модуль 3. Циклы

Тема 1. Введение

Краткое описание целей и задач, которые будут решаться на протяжении всего модуля.

Тема 2. Цикл for

Цикл for. Совмещённые операторы.

Тема 3. Цикл while

Цикл while.

Тема 4. Работа с вложенными циклами

Вложенные циклы.

Тема 5. Дополнительные возможности

Enumerate. Break. Continue. Практическое использование циклов для задач обработки текстов.

Тема 6. Закрепление знаний

Практика.

Модуль 4.1 Введение в функции в Python

Тема 1. Введение

Краткое описание целей и задач, которые будут решаться на протяжении всего модуля.

Тема 2. Функции в Python

Сигнатура и семантика функции. Оператор return. Функции как инструмент проектирования структуры программы.

Тема 3. Проверка аргументов. Аргументы по умолчанию

Проверка аргументов на корректность. Аргументы по умолчанию.

Тема 4. Продвинутое передачу аргументов

Способы передачи аргументов в Python.

Тема 5. Продвинутое обработка аргументов

Обработка неизвестного заранее числа аргументов в Python.

Тема 6. Lambda-функции

Lambda-функции.

Тема 7. Закрепление знаний

Часть I. Часть II.

Отработка знаний на практических задачах.

Модуль 4.2 Продвинутое использование функций в Python

Тема 1. Вложенные функции. Область видимости переменных

Вложенные функции. Разрешение переменных. Области видимости. Иллюстрация работы правила legb.

Тема 2. Изменение переменных вне области видимости

Изменение значений глобальных переменных. Объявление `global`. Изменение значений нелокальных переменных. Объявление `nonlocal`. Изменение значений встроенных переменных.

Тема 3. Вложенные функции и области видимости. Практика

Тестовые задания на понимание области видимости. Задача о регистрации пользователей. Задача о геометрических фигурах.

Тема 4. Рекурсия

Определение рекурсии. Сумма элементов списка через рекурсию. Стек вызова функций. Факториал через рекурсию. Глубина рекурсии. Цикл vs рекурсия.

Тема 5. Рекурсия. Практика

Тестирование по теме рекурсия. Расставить скобки. Работа со вложенными списками. Древовидные структуры.

Тема 6. Встроенная функция `map()`

Функция как объект. Функция `map`. `Map` и пользовательские функции. `Map+lambda`.

Тема 7. Встроенная функция `filter()`

Функция `filter`. `filter + lambda`. Конвейеры из `map` и `filter`.

Дополнительный модуль. Принципы ООП в Python и отладке кода

Тема 1. Введение

Краткое описание целей и задач, которые будут решаться на протяжении всего модуля.

Тема 2. Принципы ООП

Наследование. Абстракция. Инкапсуляция. Полиморфизм.

Тема 3. Объекты и классы

Объекты. Классы. Объекты из классов.

Тема 4. Атрибуты и методы

Атрибуты и методы. Метод `_init_`. Краткое резюме.

Тема 5. Практические примеры

Отслеживание состояния. Комбинация операций. Класс-обёртка. Импорт и организация кода.

Тема 6. Проверка знаний

Задания на отработку полученных знаний, умений и навыков.

Тема 7. Применение ООП для работы с файлами

Путь к файлу. Работа с файлами. чтение и запись построчно. Файл как итератор. Менеджер контекста `with`.

Тема 8. Исключения

Исключительные ситуации или же исключения в языке `python`.

Тема 9. Тонкости обработки исключений. Собственные классы исключений

Тонкости обработки исключений. Собственные классы исключений.

Модуль 5. Основы Python. Практика

Тема 1. Введение

Краткое описание целей и задач, которые будут решаться на протяжении всего модуля.

Тема 2. Основы Python

Задания на отработку полученных знаний, умений и навыков.

Тема 3. Погружение в типы данных

Задания на отработку полученных знаний, умений и навыков.

Тема 4. Условные операторы

Задания на отработку полученных знаний, умений и навыков.

Тема 5. Циклы

Задания на отработку полученных знаний, умений и навыков.

Тема 6. Функции

Задания на отработку полученных знаний, умений и навыков.

Тема 7. Тематический проект

Задания на отработку полученных знаний, умений и навыков.

Тема 8. Задачи с собеседований

Задания на отработку полученных знаний, умений и навыков.

Дополнительный модуль. Как писать красивый код на Python?

Тема 1. Зачем нужен стандарт PEP 8?

Почему нам нужен pep 8?

Тема 2. Названия объектов в Python

Общие рекомендации.

Тема 3. Макет кода

Как обрабатывать ограничение в 79 символов, рекомендованное в pep 8.

Тема 4. Комментарии

Использование комментариев.

Тема 5. Пробелы около бинарных операторов

Задания на отработку полученных знаний, умений и навыков.

Тема 6. Скринкаст: оформление функции по стандарту PEP 8

Просмотр скринкаста.

Модуль 6.1 Инструменты для Data Science.

Тема 1. Введение

Краткое описание целей и задач, которые будут решаться на протяжении всего модуля.

Тема 2. Интегрированная среда разработки (IDE)

Что такое интегрированная среда разработки? Преимущества VC Code. Установка Python и VC Code. Начало работы с VC Code.

Тема 3. Базовые функции VS Code

Обзор функционала vs code. Приложения и расширения для vs code. Установка модуля numpy. Работа с кодом. Дебаггинг.

Тема 4. Jupyter Notebook

Работа в Jupyter Notebook. Jupyter как отдельное приложение.

Тема 5. Работа с GitHub

Установка и настройка GitHub. Создание репозитория на GitHub. Создание локального репозитория и commit. Синхронизация репозитория и публикация. Объединение веток. Оформление проекта на GitHub.

Тема 6. Воспроизводимость кода

Воспроизводимость кода.

Тема 7. Работа с Google Colab

Google Colab — ещё один полезный для работы сервис.

Модуль 6.2 Библиотека NumPy

Тема 1. Введение

Краткое описание целей и задач, которые будут решаться на протяжении всего модуля.

Тема 2. Модуль Collections. Counter и defaultdict

Учимся подсчитывать элементы в списке с помощью объекта Counter и создавать словарь с заданным по умолчанию объектом для упрощения написания кода.

Тема 3. Модуль Collections. Deque и OrderedDict

Знакомимся с инструментом deque, который позволяет работать с очередью элементов, и узнаете о таких структурах данных, как очередь и рюкзак (он же стек).

Освоим ещё один словарь — OrderedDict. Он гарантирует сохранение ключей в том порядке, в котором их добавляли в словарь.

Тема 4. Модуль Collections. Закрепление знаний

Закрепление знаний на практике.

Тема 5. Модуль NumPy. Типы данных

Установка и импорт библиотеки NumPy. Использование встроенных в NumPy типов данных. Преобразование одних типов данных в другие.

Тема 6. Модуль NumPy. Массивы

Создание одномерных и многомерных массивов. Заполнение массивов числами. Изменение формы массивов. Преобразование типов данных в массиве.

Тема 7. Модуль NumPy. Действия с массивами

Изменение формы массива. Получение элемента из массива по индексу. Получение срезов массивов. Сортировка одномерных массивов. работа с пропущенными значениями.

Тема 8. Модуль NumPy. Операции с векторами
Учимся производить арифметические операции с векторами. Считать скалярное произведение векторов и расстояние между ними. Получать базовую статистику о векторе.

Тема 9. Модуль NumPy. Случайные числа
Генерация наборов случайных чисел в NumPy. Перемешивание элементов в массивах. Получение одинаковых наборов случайных чисел с помощью seed.

Тема 10. Модуль NumPy. Закрепление знаний
Закрепление знаний на практике.

Дополнительный модуль. Как выгружать данные из файлов разных форматов

Тема 1. Введение
Краткое описание целей и задач, которые будут решаться на протяжении всего модуля.

Тема 2. Работа с текстовыми файлами
Используем функцию read_table(). Применение параметра header. Решаем проблему с кодировкой исходных данных. Чтение файла по ссылке, используя функцию read_table(). Чтение файла по ссылке, используя функцию read_table().

Тема 3. Работа с файлами Excel
Немного об Excel. Считывание данных из файла Excel. Считывание данных из файла Excel по ссылке. Выгрузка данных из dataframe в Excel-файл.

Тема 4. JSON. Что это?
Json. модули для работы с JSON. Как выглядит JSON-файл?

Тема 5. JSON. Открываем JSON-файл и извлекаем данные
Открываем JSON-файл. Извлекаем данные из JSON-файла.

Тема 6. JSON. Работаем с pandas. Из JSON в pandas
Из JSON в pandas. Сохраняем dataframe в csv-файле.

Тема 7. JSON. Работаем с pandas. Из pandas в JSON
Из pandas в JSON.

Тема 8. XML. Что это?
Расширяемый язык разметки.

Тема 9. Контент XML-файла
Извлекаем контент из XML-файла. Корень. Потомки. атрибуты и теги. Атрибуты и теги.

Тема 10. XML. Загружаем, создаем, сохраняем
Загружаем данные из XML-файла в dataframe. Создаём XML-файл. Сохранение XML-файла.

Дополнительный модуль. Как получать данные из веб-источников и API

Тема 1. Введение
Краткое описание целей и задач, которые будут решаться на протяжении всего модуля.

Тема 2. Веб-запросы
Методы запросов в протоколе HTTP.
Тема 3. Библиотека requests
Работаем с ответом.

Тема 4. Парсинг сайтов
Основы HTML. Получаем содержимое веб-страницы.

Тема 5. Библиотека BeautifulSoup
Извлекаем заголовок и время написания статьи. Неуникальные теги: извлекаем текст и дату публикации статьи. Сбор нескольких элементов: собираем все ссылки на странице.

Тема 6. Работа с API
Ключ авторизации. первые запросы к API. запрос к API из кода. Сбор информации из групп. Ограничение по частоте запросов. Лайки, репосты и комментарии. Другие API.

Тема 7. Как настроить регулярную выгрузку данных
Скрипты. как настроить автоматический запуск. Постановка задачи. Выполнение функции.

Модуль 7.1 Введение в Pandas

Тема 1. Введение
Краткое описание целей и задач, которые будут решаться на протяжении всего модуля.

Тема 2. Pandas.Series
Series как структура данных. Создание Series. Доступ к данным в Series.

Тема 3. Pandas.DataFrame
DataFrame как структура данных. Создание DataFrame. Axis в DataFrame. Доступ к данным в DataFrame.

Тема 4. Работа с различными источниками данных в Pandas
Запись в csv-файл. Чтение csv-файла. Чтение csv-файла по ссылке. Запись и чтение в других форматах.

Тема 5. Знакомимся с данными: недвижимость
Разбор инструментария pandas на наборе данных с учебного соревнования по анализу данных на kaggle.

Тема 6. Исследование структуры DataFrame
Вывод первых и последних строк. Размерность таблицы. Получение информации о столбцах. Изменение типа данных в столбце. получение описательной статистики. Получение частоты уникальных значений в столбце.

Тема 7. Статистические методы
Агрегирующие методы. Модальное значение.

Тема 8. Фильтрация данных в DataFrame
Фильтрация с помощью масок.

Тема 9. Закрепление знаний

Задания на отработку полученных знаний, умений и навыков.

Модуль 7.2 Базовые приемы работы с данными в Pandas

Тема 1. Базовые операции со столбцами DataFrame

Создание копии таблицы. Удаление столбцов. Математические операции со столбцами.

Тема 2. Работа с датами в DataFrame

Признаки даты и времени. Формат datetime. Выделение атрибутов datetime. Работа с интервалами.

Тема 3. Создание и преобразование столбцов с помощью функций

Сложные манипуляции над столбцами.

Тема 4. Тип данных Category

Признаки: категориальные и числовые. Категории в данных о недвижимости. Тип данных Category.

Модуль 8. Продвинутое методы работы с данными в Pandas

Тема 1. Введение

Краткое описание целей и задач, которые будут решаться на протяжении всего модуля.

Тема 2. Сортировка данных в DataFrame

Метод `sort_values()`.

Тема 3. Группировка данных в DataFrame

Метод `groupby()`.

Тема 4. Сводные таблицы

Метод `groupby` как способ построения сводных таблиц. Метод `pivot_table` для построения сводных таблиц. Многомерные сводные таблицы. Доступ к данным в сводной таблице.

Тема 5. Объединение DataFrame: знакомимся с новыми данными

С какими данными мы работаем? Зачем хранить данные в разных таблицах?

Тема 6. Объединение DataFrame: `concat`

Склеивание (конкатенация) таблицы как по строкам, так и по столбцам.

Тема 7. Объединение DataFrame: `join`, `merge`

Типы объединений. Метод объединения `join`. Метод объединения `merge`. Какой метод объединения использовать?

Модуль 9.1 Визуализация данных.

Тема 1. Введение

Краткое описание целей и задач, которые будут решаться на протяжении всего модуля.

Тема 2. Обзор типов визуализации

Обзор типов диаграмм.

Тема 3. Знакомимся с новыми данными: коронавирус
Предобработка данных.

Тема 4. Графические возможности библиотеки Pandas
Базовая визуализация в pandas.

Тема 5. Графические возможности библиотеки Matplotlib
Немного о библиотеке. Основные объекты matplotlib. Добавление информативности в графики. Использование нескольких систем координат. Subplots.

Тема 6. Графические возможности библиотеки Seaborn
Немного о библиотеке.

Тема 7. Графические возможности библиотеки Plotly
Немного о библиотеке. экспресс-режим. Анимация графиков во времени.
Трёхмерная визуализация. сохранение графика plotly.

Тема 8. Искусство визуализации
Общие рекомендации к созданию визуализации. Рекомендации по использованию популярных типов диаграмм.

Модуль 9.2 Очистка данных.

Тема 1. Знакомство с новыми данными: данные о квартирах от Сбера
Работа с данными с самого настоящего соревнования на платформе Kaggle.

Тема 2. Работа с пропусками: как их обнаружить?
Причины появления пропусков в данных. Как обнаружить пропуски? Список столбцов с пропусками. Столбчатая диаграмма пропусков. Тепловая карта пропусков.

Тема 3. Работа с пропусками: методы обработки
Методы обработки пропущенных значений. Другие методы заполнения отсутствующих значений.

Тема 4. Выбросы: почему появляются и чем опасны?
Чем опасны выбросы? Причины появления выбросов.

Тема 5. Методы выявления выбросов
Метод ручного поиска и здравого смысла. Метод межквартильного размаха (метод Тьюки). Другие методы поиска выбросов.

Тема 6. Работа с дубликатами и неинформативными признаками
Дубликаты. чем опасны дубликаты? Неинформативные признаки.

Дополнительный модуль. Markdown и GIT для создания портфолио

Тема 1. Введение
Краткое описание целей и задач, которые будут решаться на протяжении всего модуля.

Тема 2. Язык разметки Markdown
Что такое язык разметки? Язык разметки Markdown. Редакторы Markdown.

Тема 3. Синтаксис Markdown

Шрифт. Заголовки. Списки. Ссылки и изображения. Программный код и цитирование. Формулы. Пример оформления файла readme.md.

Тема 4. Системы контроля версий. Git и GitHub.

Зачем управлять версиями? Типы систем контроля версий. Система контроля версий git. GitHub — хостинг it-проектов. Создание удалённого репозитория. Создаём удалённый репозиторий проекта.

Тема 5. Git. Основные операции

Получение данных о состоянии репозитория. Перемещение между коммитами и откат изменений.

Тема 6. Git. Игнорирование и работа с удалённым репозиторием

Файл .gitignore. Работа с удалённым репозиторием.

Тема 7. Git. Ветвление и конфликты

Что такое ветвление? Создание ветки и переключение на ветку. Просмотр списка веток и их состояний. Слияние веток. Конфликты.

Тема 8. Методологии ветвления. Культура коммитов. Форк

Культура коммитов. Форк.

Тема 9. Рекомендации к составлению портфолио на GitHub

Оформление репозитория. Кастомизация стартовой страницы на GitHub. Что показать, если показать нечего.

9. Текущий контроль по дисциплине

Текущий контроль по дисциплине проводится путем проведения контрольных работ, тестов по лекционному материалу, выполнения домашних заданий и проектов, и фиксируется в форме контрольной точки не менее одного раза в семестр. Оценочные материалы текущего контроля размещены на сайте ТГУ в разделе «Информация об образовательной программе» – <https://www.tsu.ru/sveden/education/eduop/>.

10. Порядок проведения и критерии оценивания промежуточной аттестации

Промежуточный контроль осуществляется в виде тестирования. Время проведения письменного экзамена составляет 2 академических часа. Во время проведения экзамена обучающиеся могут пользоваться программой дисциплины. Вопросы и задания экзамена покрывают основные темы, изученные в рамках дисциплины:

- основы и синтаксис Python
- типы данных
- условные операторы, циклы, функции
- основные библиотеки
- очистка и визуализация данных

Результаты экзамена определяются оценками «отлично», «хорошо», «удовлетворительно», «неудовлетворительно».

Оценочные материалы для проведения промежуточной аттестации размещены на сайте ТГУ в разделе «Информация об образовательной программе» – <https://www.tsu.ru/sveden/education/eduop/>.

11. Учебно-методическое обеспечение

- а) Электронный учебный курс по дисциплине на интерактивной платформе «Skillfactory CS».
- б) Оценочные материалы текущего контроля и промежуточной аттестации по дисциплине (<https://www.tsu.ru/sveden/education/eduop/>).
- в) План семинарских / практических занятий по дисциплине.
- г) Методические указания по организации самостоятельной работы студентов.

12. Перечень учебной литературы и ресурсов сети Интернет

- а) основная литература:
 - Сузи Р.А. Язык программирования Python [Электронный ресурс]/ Сузи Р.А.— Электрон. текстовые данные.— Москва: Интернет-Университет Информационных Технологий (ИНТУИТ), 2016.— 350 с.
 - Васильев А.Н. Python на примерах [Электронный ресурс]: практический курс по программированию/ Васильев А.Н.— Электрон. текстовые данные.— Санкт-Петербург: Наука и Техника, 2017.— 432 с.
 - Маккинли Уэс Python и анализ данных [Электронный ресурс]/ Маккинли Уэс— Электрон. текстовые данные.— Саратов: Профобразование, 2019.— 482 с.
- б) дополнительная литература:
 - Горожанина Е.И. Нейронные сети [Электронный ресурс]: учебное пособие/ Горожанина Е.И.— Электрон. текстовые данные.— Самара Поволжский государственный университет телекоммуникаций и информатики, 2017.— 84 с.— Режим доступа: <http://www.iprbookshop.ru/75391.html>.— ЭБС «IPRbooks».
 - Павлова А.И. Информационные технологии: основные положения теории искусственных нейронных сетей [Электронный ресурс]: учебное пособие/ Павлова А.И.— Электрон. текстовые данные.— Новосибирск: Новосибирский государственный университет экономики и управления «НИНХ», 2017.— 191 с.— Режим доступа: <http://www.iprbookshop.ru/87110.html>.— ЭБС «IPRbooks».
- в) ресурсы сети Интернет:
 - открытые онлайн-курсы;

13. Перечень информационных технологий

- а) лицензионное и свободно распространяемое программное обеспечение:
 - Microsoft Office Standart 2013 Russian: пакет программ. Включает приложения: MS Office Word, MS Office Excel, MS Office PowerPoint, MS Office OneNote, MS Office Publisher, MS Outlook, MS Office Web Apps (Word Excel MS PowerPoint Outlook);
 - публично доступные облачные технологии (Google Docs, Яндекс диск и т.п.).
- б) информационные справочные системы:
 - Электронный каталог Научной библиотеки ТГУ – <http://chamo.lib.tsu.ru/search/query?locale=ru&theme=system>
 - Электронная библиотека (репозиторий) ТГУ – <http://vital.lib.tsu.ru/vital/access/manager/Index>
 - ЭБС Лань – <http://e.lanbook.com/>
 - ЭБС Консультант студента – <http://www.studentlibrary.ru/>
 - Образовательная платформа Юрайт – <https://urait.ru/>
 - ЭБС ZNANIUM.com – <https://znanium.com/>

– ЭБС IPRbooks – <http://www.iprbookshop.ru/>

14. Материально-техническое обеспечение

Каждый обучающийся обеспечен доступом к интерактивной платформе «Skillfactory CS» <https://apps.skillfactory.ru/learning/login>.

15. Информация о разработчиках

Кузовкова Александра Сергеевна, методист Skillfactory