15 проектов на Python для начинающих, чтобы практиковаться и учиться

Описание

В этой статье мы рассмотрим 15 проектов Python с кодом, которые помогут вам создать портфолио и повысить уровень своих навыков программирования на Python. Мы начнем с обсуждения того, почему вы должны изучать Python, а затем поможем вам выбрать среду разработки. Попутно мы также поможем вам советами по эффективному изучению Python. Затем мы проведем вас по списку из 15 проектов на Python для начинающих, охватывающих широкий спектр тем. Чтобы узнать все это и многое другое, давайте начнем!

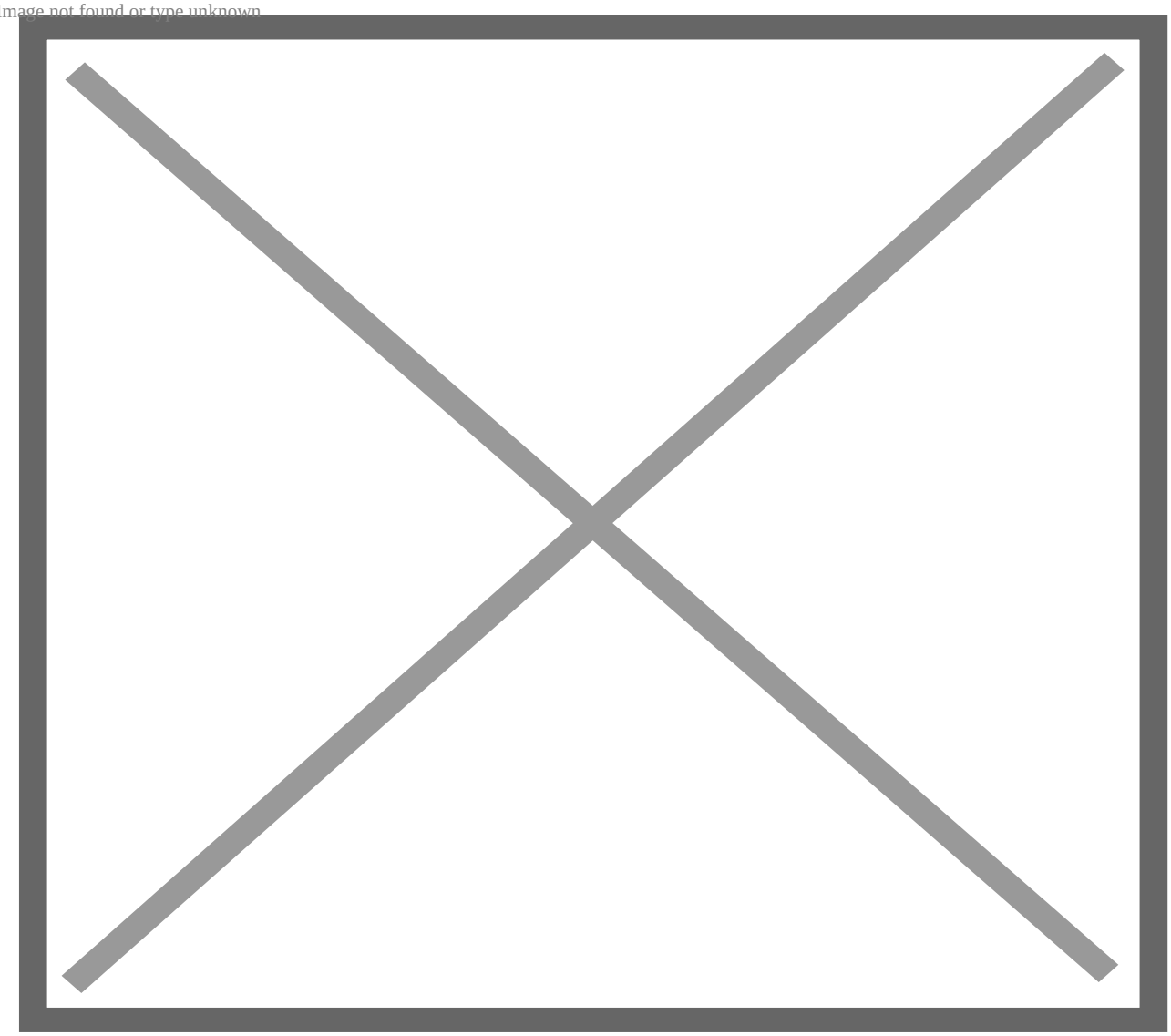
Почему вы должны изучать Python?

Легко читаемый синтаксис, множество встроенных модулей и дружное сообщество делают Python отличным первым языком программирования для изучения. Как только вы познакомитесь с основами программирования, вы, возможно, захотите выбрать специализацию и освоить конкретные навыки. Ознакомьтесь со следующим списком специализаций. Заинтересовала ли вас какая-нибудь из них?

- Веб-разработка
- Разработка API
- Аналитика данных
- Наука о данных
- Инженерия данных
- Автоматизация

- Разработка игр

Image not found or type unknown



Это одни из самых востребованных направлений; вы можете делать все это и многое другое с помощью Python! Так что если вы хотите изучать Python, лучшее время начать – сейчас!

Какие лучшие IDE и редакторы кода Python можно использовать?

Если вы начинающий программист Python, вам, скорее всего, понадобится помощь в выборе лучшей среды разработки. Для решения проблем кодирования и создания простых модулей достаточно редактора кода. Когда вы начнете создавать более

крупные проекты с множеством модулей, вам, возможно, захочется рассмотреть интегрированную среду разработки (IDE). IDE поможет вам управлять установками, разрабатывать и отлаживать в одной среде. Вот наш список лучших редакторов кода Python и IDE.

Sublime Text

Благодаря дружественному интерфейсу, поддержке Markdown и множества языков программирования, Sublime Text является одним из лучших редакторов кода, который можно использовать для написания сценариев на Python.

Geekflare Онлайн-редактор кода Python

У нас есть библиотека учебных пособий по Python, и вы можете открыть онлайн-редактор Python от Geekflare и попробовать выполнить фрагменты кода, проходя учебники. Вы можете сделать это прямо из браузера – установка не требуется!

Jupyter Notebook

Любите ли вы, помимо написания кода, добавлять описательный текст и изображения для документирования своего проекта? Если да, то Jupyter Notebook может стать отличным выбором для демонстрации ваших проектов! Помимо традиционного интерфейса блокнота, существует несколько альтернатив Jupyter, которые обеспечивают лучшую совместную работу и расширенные возможности.

PyCharm

PyCharm от JetBrains – еще одна популярная IDE для Python. Вы можете управлять своими проектами в виртуальных средах, устанавливать необходимые пакеты и профилировать код Python. Существует бесплатная версия для сообщества, которую вы можете использовать.

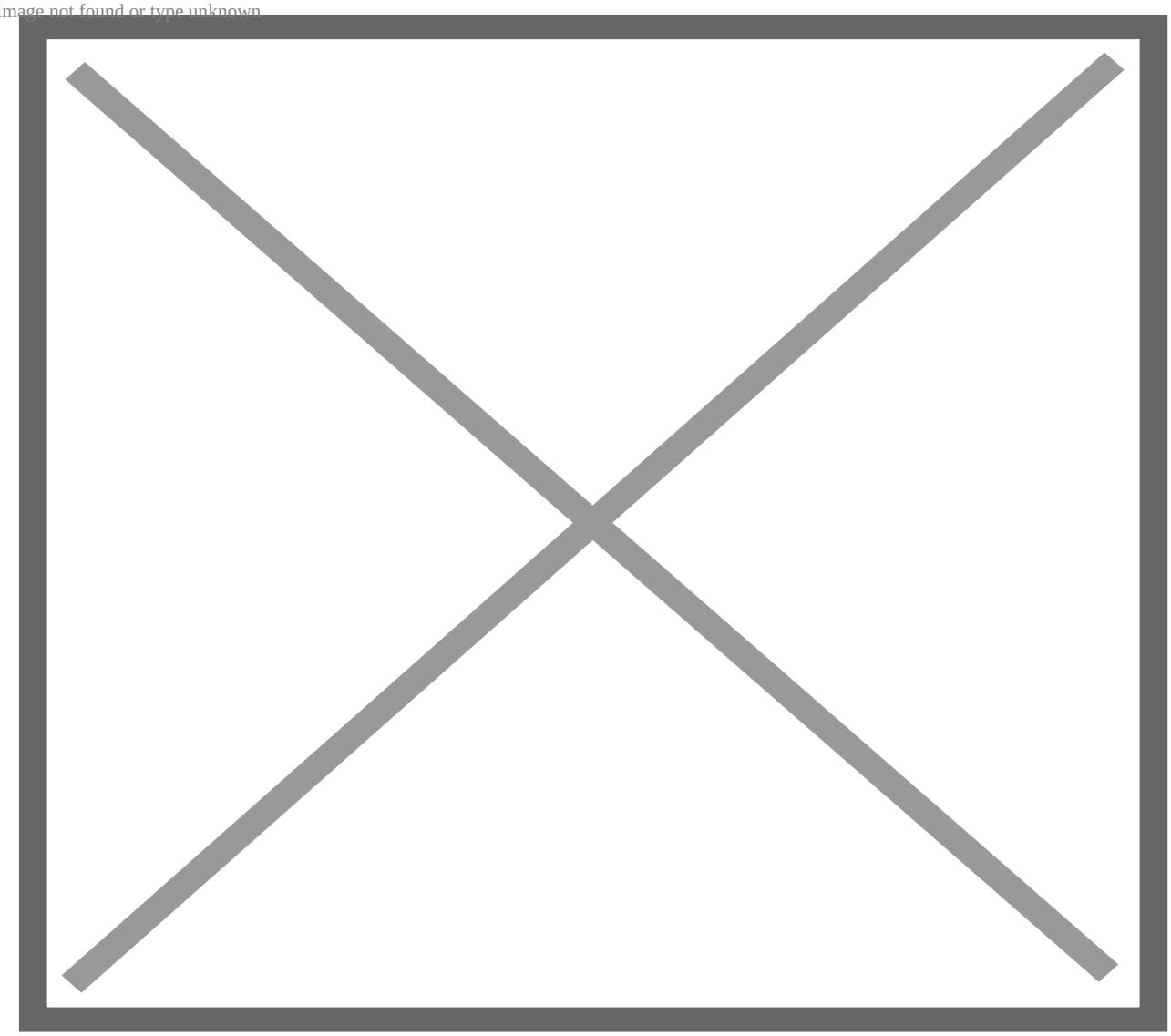
Visual Studio Code

Visual Studio Code, или VS Code от Microsoft – еще одна бесплатная IDE, популярная в сообществе разработчиков. VS Code имеет расширения, которые поддерживают использование широкого спектра технологических стеков и языков программирования, таких как HTML, CSS, JavaScript, Go и Python.

Как лучше всего изучать Python?

Теперь, когда у вас есть список вариантов для разработки и программирования, давайте посмотрим, как вы можете эффективно изучать Python. Существует три этапа эффективного изучения любого языка программирования.

Image not found or type unknown



Шаг 1: Изучите основы

Сюда входит изучение базового синтаксиса, типов данных и встроенных структур данных, основ программирования, таких как условные операторы, циклы, функции и так далее. Вы можете изучать основы Python онлайн на таких платформах, как Udemy.

Шаг 2: Создание проектов

Изучение синтаксиса – это только вершина айсберга, когда речь идет об освоении языка программирования. Настоящее обучение происходит, когда вы создаете проекты, применяя полученные знания.

Шаг 3: Практика

На данном этапе вы поняли основы, а также создали проекты. Как сделать так, чтобы вы не забывали обо всем, чему научились? Ключевым моментом является целенаправленная практика. Это означает, что вы должны периодически пересматривать и практиковаться, а также напрягать свои кодовые мышцы.

Проекты на Python для начинающих

Поскольку создание проектов и решение практических задач является ключом к эффективному обучению, мы составили список из 15 проектов и вопросов по кодированию на Python для начинающих – все они объясняются с помощью кода. Эти практические задачи и проекты начнутся с изучения основ. Затем список постепенно познакомит вас с различными модулями Python и парадигмами программирования, такими как объектно-ориентированное программирование (ООП), чтобы помочь вам пройти путь от начинающего до среднего программиста на Python. Давайте начнем.

Решение задач на строках Python

Как начинающий программист Python, вы должны быть знакомы со строками Python и строковыми операциями. Чтобы помочь вам повысить уровень знаний, мы подготовили учебник, в котором рассматриваются интересные проблемы,связанные со строками Python.

Проблемы, которые вы будете решать:

- Проверьте, является ли данная строка Python палиндромом.
- Проверьте, являются ли две строки анаграммами.
- Если дана строка, содержащая имя и фамилию, проверьте, отформатирована ли строка имени в заглавном регистре.

Попутно вы также изучите такие важные понятия, как нарезка строк и разворот строк в Python.

Работа с датами и временем в Python

Еще один важный модуль в ядре Python – datetime. В Python есть встроенный модуль дата-время, который помогает эффективно работать с датами и временем. Мы подготовили руководство по работе с датами и временем в Python с акцентом на вычисление разницы во времени.

Что вы узнаете:

- Работа с объектами date, time и datetime в Python
- Использование объекта timedelta в Python для вычисления разницы во времени

Что вы будете решать:

1. Получение текущей даты и времени в Python
2. Вычислить время, оставшееся до вашего дня рождения
3. Вычислите количество дней, оставшихся до Дня учителя, Рождества и других праздников в этом году.

Проверка правильности числа в Python

В качестве следующего шага вы можете изучить учебники по Python, посвященные основам математики и логики. Вот популярный вопрос: Как проверить, является ли число простым в Python?

Что вы будете программировать:

- O(n) алгоритм проверки, является ли число простым или нет.
- O(√n) алгоритм для проверки простого числа – используя тот факт, что факторы числа встречаются попарно.

Умножение матриц в Python

Матрицы и вопросы о матрицах часто всплывают на собеседованиях по кодингу. Чтобы помочь вам начать работу, мы подготовили подробное руководство по умножению матриц в Python.

Что вы узнаете:

Кодирование пользовательской функции Python для умножения матриц
Использование понимания вложенного списка для выполнения умножения матриц
Использование встроенных функций NumPy для умножения матриц

Работа с текстовыми файлами в Python

В любом языке программирования работа с файлами является важным аспектом. И мы подготовили подробное руководство по работе с файлами в Python.

Что вы узнаете:

- Работа с текстовыми файлами в Python
- Общие операции ввода-вывода файлов
- Работа с данными в файлах

Печать треугольника Паскаля в Python

Теперь, когда мы сосредоточились на математических задачах, печать треугольника Паскаля – часто задаваемый вопрос на технических собеседованиях.

Что вы узнаете:

- Как построить треугольник Паскаля
- Как реализовать функцию Python для печати треугольника Паскаля для любого заданного количества строк.

Проверка допустимых круглых скобок в Python

Проверка правильности строки с круглыми скобками – популярный вопрос на собеседовании, и у нас есть учебник, который поможет вам его решить!

Что вы узнаете:

- Использовать встроенные структуры данных Python: списки и словари для эффективного решения задачи.
- Проверять на достоверность любую заданную строку со скобками.

После изучения основ Python важно освоить базовые алгоритмы поиска и сортировки, и мы подготовили для вас исчерпывающие учебные пособия по этим вопросам.

Алгоритмы поиска в Python

Если вы хотите найти определенный элемент в массиве, вам помогут алгоритмы поиска.

- Линейный поиск со сложностью O(n), для входного массива размера n
- Бинарный поиск со сложностью O(log n)

Если линейный поиск – это лучшее, что вы можете сделать, когда входной массив не отсортирован, то двоичный поиск наиболее эффективен, когда у вас есть отсортированный массив. Это связано с тем, что двоичный поиск работает путем уменьшения интервала поиска в два раза на каждом последующем шаге.

Алгоритмы сортировки в Python

После того, как вы освоите поиск по спискам, полезно узнать, как сортировать списки. У нас есть учебник, который научит вас кодировать некоторые из основных алгоритмов сортировки в Python.

Что вы узнаете:

- Сортировка вставкой
- Сортировка выбором
- Пузырьковая сортировка
- Сортировка слиянием

Создание игры для угадывания чисел

Игра на угадывание чисел, в которой вы пытаетесь угадать секретное число за минимальное количество угадываний, – это забавный проект по кодингу.

Что вы будете кодировать:

- Проверять вводимые пользователем данные на соответствие секретному числу
- Давать пользователям обратную связь по их угадыванию, а также ограничивать максимально допустимое количество угадываний

Создание генератора паролей на Python

Вам когда-нибудь было трудно придумывать надежные пароли для всех сайтов, которые вы часто посещаете в Интернете? Как насчет того, чтобы написать свой собственный генератор паролей на Python? Вы можете использовать Python для генерации паролей, отвечающих таким ограничениям, как количество специальных символов, длина пароля и многое другое.

Что вы будете создавать:

- Введя количество символов каждого типа, сгенерировать надежный пароль, удовлетворяющий всем ограничениям.
- Эффективно использовать модуль Python secrets

Отправка электронных писем с помощью Python

Вот простой проект автоматизации на языке Python. Вы можете использовать Python для автоматизации отправки электронных писем. Поздравить коллег с особым событием, а друзей с днем рождения еще никогда не было так просто!

Чему вы научитесь:

- Автоматизировать отправку электронных писем
- Использовать smtplib, реализацию протокола Simple Mail Transfer Protocol на языке Python

Создание игры “Крестики-нолики” на Python

Крестики-нолики – это одна из игр, в которую мы все играли с нашими друзьями в какой-то момент времени. Разве не было бы здорово, если бы вы могли написать ее на Python и пригласить друзей поиграть в нее? Да, и у нас есть учебник, который поможет вам создать свою собственную игру “Крестики-нолики”. Вы переведете логику игры в крестики-нолики в код Python.

Что вы будете использовать:

- Основы объектно-ориентированного программирования (ООП)
- Понимание методов классов

Создание приложения для работы с таблицей умножения

Python – это язык программирования общего назначения, который поддерживает как функциональную, так и объектно-ориентированную парадигму программирования (ООП). Чтобы помочь вам улучшить свои навыки ООП, мы подготовили для вас проект.

Что вы будете создавать:

- Приложение для работы с таблицей умножения с использованием ООП
- Научитесь использовать методы и атрибуты классов

Код игры “Змейка” на Python

Помните классическую игру в змейку? Используя все полученные знания, вы создадите игру-змейку на Python с графическим интерфейсом пользователя (GUI), переведя логику игры в код Python.

Что вы создадите:

- Простой графический интерфейс пользователя на Python с использованием библиотеки turtle.
- Сформулируете правила игры в змейку и настроите игровую среду.

Подведение итогов

Надеюсь, вы нашли несколько интересных практических задач и проектов, которые можно добавить в свое портфолио. По мере того, как вы будете добавлять новые навыки в свой инструментарий Python, не забывайте дополнять свое обучение созданием проектов. Помните, что постоянная практика – это ключ к совершенствованию в программировании. До скорой встречи в очередном уроке по Python! Счастливого обучения!

Дата Создания
28.06.2023