|  |  |
| --- | --- |
|  | **Министерство науки и высшего образования Российской Федерации**  Калужский филиал  федерального государственного бюджетного  образовательного учреждения высшего образования  ***«Московский государственный технический университет имени Н.Э. Баумана (национальный исследовательский университет)»***  ***(КФ МГТУ им. Н.Э. Баумана)*** |

**ФАКУЛЬТЕТ** ***ИУ-КФ «Информатика и управление» \_\_\_\_\_\_\_\_\_***

**КАФЕДРА** ***ИУ5-КФ «Системы обработки информации»***

**РАСЧЕТНО-ПОЯСНИТЕЛЬНАЯ ЗАПИСКА**

**к курсовому проекту на тему:**

***\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_***

по дисциплине ***Базы данных***

Студент гр.\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ (\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_)

(подпись) (Ф.И.О.)

Руководитель \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ (\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_)

(подпись) (Ф.И.О.)

Оценка руководителя \_\_\_\_\_ баллов \_\_\_\_\_\_\_\_\_\_\_

30-50 (дата)

Оценка защиты \_\_\_\_\_ баллов \_\_\_\_\_\_\_\_\_\_\_

30-50 (дата)

Оценка проекта \_\_\_\_\_ баллов \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

(оценка по пятибалльной шкале)

Комиссия: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ (\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_)

(подпись) (Ф.И.О.)

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ (\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_)

(подпись) (Ф.И.О.)

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ (\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_)

(подпись) (Ф.И.О.)

Калуга, 2020

Калужский филиал   
федерального государственного бюджетного образовательного учреждения высшего образования

***«Московский государственный технический университет имени Н.Э. Баумана (национальный исследовательский университет)»   
(КФ МГТУ им. Н.Э. Баумана)***

УТВЕРЖДАЮ

Заведующий кафедрой **\_\_ИУ5-КФ\_\_\_**

\_\_\_\_\_\_\_\_\_\_\_(\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_)

«\_\_\_\_»\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_20\_\_\_г.

**ЗАДАНИЕ**

**на выполнение курсового проекта**

по дисциплине ***Базы данных***

Студент *Карачёнков Э.В., СОИ.Б-51*

(фамилия, инициалы, индекс группы)

Руководитель\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ *Кириллов В.Ю.*

(фамилия, инициалы)

График выполнения проекта: 25% к 4 нед., 50% к 7 нед., 75% к 10 нед., 100% к 14 нед.

***1. Тема курсового проекта***

***\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_***

***2. Техническое задание***

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

***3. Оформление курсового проекта***

3.1. Расчетно-пояснительная записка на\_\_\_\_\_\_\_\_ листах формата А4.

3.2. Перечень графического материала КП (плакаты, схемы, чертежи и т.п.) \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Дата выдачи задания «\_\_\_\_\_»\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_20\_\_г.

Руководитель курсового проекта \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_/\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_/

(подпись) (Ф.И.О.)

Задание получил\_\_\_\_\_\_\_\_\_\_\_\_\_\_/\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_/ «\_\_\_\_\_»\_\_\_\_\_\_\_\_\_\_20\_\_г.

(подпись) (Ф.И.О.)

Примечание: Задание оформляется в двух экземплярах: один выдается студенту, второй хранится на кафедре.

**Оглавление**

[1. Техническое задание 6](#_Toc55737655)

[1.1 Наименование 6](#_Toc55737656)

[1.2 Основания для разработки 6](#_Toc55737657)

[1.3 Исполнитель 6](#_Toc55737658)

[1.4 Цель разработки 6](#_Toc55737659)

[1.5 Содержание работы 6](#_Toc55737660)

[1.5.1 Задачи, подлежащие решению 6](#_Toc55737661)

[1.5.2 Требования к архитектуре АСОИ 6](#_Toc55737662)

[1.5.3 Требования к составу программных компонентов 7](#_Toc55737663)

[1.5.4 Требования к прикладным программам 7](#_Toc55737664)

[1.5.5 Требования к входным/выходным данным 7](#_Toc55737665)

[1.5.6 Требования к временным характеристикам 7](#_Toc55737666)

[1.5.7 Требования к составу технических средств 7](#_Toc55737667)

[1.6 Этапы разработки 8](#_Toc55737668)

[1.7 Техническая документация, предъявляемая по окончании работы 8](#_Toc55737669)

[1.8 Дополнительные условия 9](#_Toc55737670)

[2. ИССЛЕДОВАТЕЛЬСКАЯ ЧАСТЬ 10](#_Toc55737671)

[2.1 Постановка задачи 10](#_Toc55737672)

[2.2 Описание предметной области 10](#_Toc55737673)

[2.3 Анализ современных СУБД 12](#_Toc55737674)

[2.3.1 СУБД PostgreSQL 13](#_Toc55737675)

[2.3.2 СУБД MySQL 14](#_Toc55737676)

[2.3.3 SQLite 15](#_Toc55737677)

[2.4 Средства разработки 17](#_Toc55737678)

# Техническое задание

## Наименование

Настоящее Техническое задание определяет требования и порядок создания веб-приложения для покупки товаров в интернет магазине.

## Основания для разработки

Основанием для разработки веб приложения для покупки товаров в магазине является задание в соответствии с учебным планом дисциплины «Базы данных»

## Исполнитель

Исполнителем проекта является студент Калужского филиала МГТУ им. Н. Э. Баумана, факультета ИУ-КФ «Информатики и управления», кафедры ИУ5-КФ «Системы обработки информации», группы СОИ.Б  51, Карачёнков Эдуард Валерьевич.

## Цель разработки

Разрабатываемое приложение предназначено для демонстрации современных подходов в веб-разработке.

Целью создания системы является изучение новых возможностей работы с базами данных в сочетании с веб-программированием.

## **Содержание работы**

### Задачи, подлежащие решению

В ходе реализации программного продукта необходимо предусмотреть интерфейс веб-приложения, придумать и описать структуры данных, функции пользователей.

### Требования к архитектуре АСОИ

Архитектура программного продукта представлена на рисунке 1.

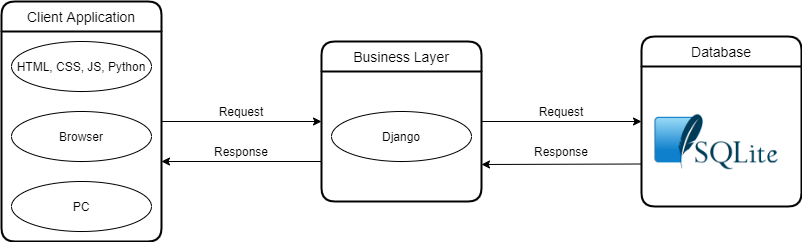


Рисунок 1  Архитектура АСОИ

В качестве основного стека технологий используется «SPBD» (SQLite, PyCharm, Bootstrap, Django).

### Требования к составу программных компонентов

Программный продукт должен представлять собой веб-приложение с правильно продуманной логикой, сервер которого находится на локальном компьютере.

### **Требования к прикладным программам**

Прикладные программы должны обеспечивать комфортную разработку для сдачи работы в срок.

### Требования к входным/выходным данным

Входные данные должны быть понятными и четкими, чтобы не нарушать логику веб-приложения.

Выходные данные должны отображать ожидаемую пользователем от приложения информацию.

### Требования к временным характеристикам

Веб-приложение должно максимально быстро работать, насколько это, конечно, возможно, чтобы не заставлять пользователя ждать.

### Требования к составу технических средств

Для построения программного продукта необходимы следующие аппаратно-технические и программные средства:

* Windows 10 x64;
* PyCharm;
* SQLite;
* Brackets.

## Этапы разработки

На этапе проектирования формулируются бизнес-цели создаваемого проекта, определяются требования, которым он должен удовлетворять, разрабатывается общая концепция. На данном этапе имеет смысл уточнить пожелания заказчика, а также произвести исследование целевой аудитории.

Разработка структуры сайта включает всё, что касается его содержания и информационной стратегии, определяющей, как должна быть организована подача информации, чтобы будущие посетители сайта могли быстро и легко её найти.

На этапе разработки программист производит объединение всех этапов и проверяет работоспособность веб-приложения. Отлавливаются баги и фиксятся проблемные моменты разработки.

## Техническая документация, предъявляемая по окончании работы

Должны быть разработаны следующие программные документы:

1. Расчетно-пояснительная записка:

* техническое задание;
* научно-исследовательская часть;
* проектирований компонентов программного продукта;
* проектно-технологическая часть.

2. Графическая часть.

## Дополнительные условия

Клиентская часть представляет собой сайт интернет-магазина, где на главной странице находится рекомендуемые товары, каталог товаров и товар дня. У каждого товара имеются название, цена, описание и количество на складе. Реализовать корзину товаров. Реализовать функцию покупки товаров из корзины. Реализовать возможность авторизации для пользователя.

Приложение разрабатывается на языке Python. Используемыми фреймворками будут Django для языка Python и Bootstrap для верстки.

# ИССЛЕДОВАТЕЛЬСКАЯ ЧАСТЬ

## Постановка задачи

Разработать и спроектировать интернет магазин вещей, отвечающий следующим требованиям:

1. Возможность добавления товара в корзину;
2. Возможность покупки товара;
3. Возможность делать автозакупку товаров при их отсутствии на складе.

## Описание предметной области

Предметная область – это часть реального мира, данные о которой отображаются в базе данных. В данной работе предметной областью является склад запчастей.

Сущность – любой различимый объект, информацию о котором необходимо хранить в базе данных. Атрибут – поименованная характеристика сущности. Его наименование должно быть уникальным для конкретного типа сущности, но может быть одинаковым для различного типа сущностей. Связь – ассоциирование двух или более сущностей.

Всего существует три типа связей:

1. Один к одному. Такая связь образуется, когда ключевой столбец (идентификатор) присутствует в другой таблице, в которой тоже является ключом либо свойствами столбца задана его уникальность (одно и тоже значение не может повторяться в разных строках).
2. Один ко многим. В данном типе связей одной записи первой таблицы соответствует несколько записей в другой таблице.
3. Многие ко многим. Если нескольким записям из одной таблицы соответствует несколько записей из другой таблицы, то такая связь называется «многие ко многим» и организовывается посредством связывающей таблицы.

Для предметной области склада были определена одна сущность – товар. Далее необходимо определить атрибуты этой сущности: у товара есть код товара, название, цена, количество на складе, описание и характеристика.

Представление — логическая таблица, представляющая собой поименованный запрос, который будет подставлен как подзапрос при использовании представления.

Пользовательское представление – представление БД с точки зрения пользователя. Состоит из внешнего представления БД, каждое представление – ведение предметной сущности пользователя и содержит только те сущности, атрибуты и связи, которые интересны пользователю.

Концептуальное представление – обобщающее представление БД, описывает, какие данные хранятся в БД, отображает связи между ними. Содержит полную структуру БД, отображающую требования организации. Концептуальный уровень включает в себя понятия объектов базы данных (таких как ключи, таблицы, индексы, представления) без учета их внутренней организации.

Концептуальное моделирование – это сбор, анализ и редактирование требований к данным о предметной области. Объектом моделирования является предметная область будущей системы, а результатом этого этапа является концептуальная модель предметной области, инвариантная к структуре базы данных и состоящая из сущностей, атрибутов и связей между ними.

На этапе концептуального проектирования строится концептуальная схема базы данных, которая затем преобразуется в реляционную схему. Этот процесс выполняется на основе методик, в которых достаточно четко оговорены все этапы такого преобразования.

Преобразование концептуальной схемы в реляционную проводится вручную или автоматически. В последнем случае известны два подхода:

1. подход, основанный на явном представлении концептуальной схемы как исходной информации для проектирования;

2. подход, ориентированный на построении интегрированных систем проектирования с автоматизированным созданием концептуальной схемы, на основе интервью с экспертами предметной области.

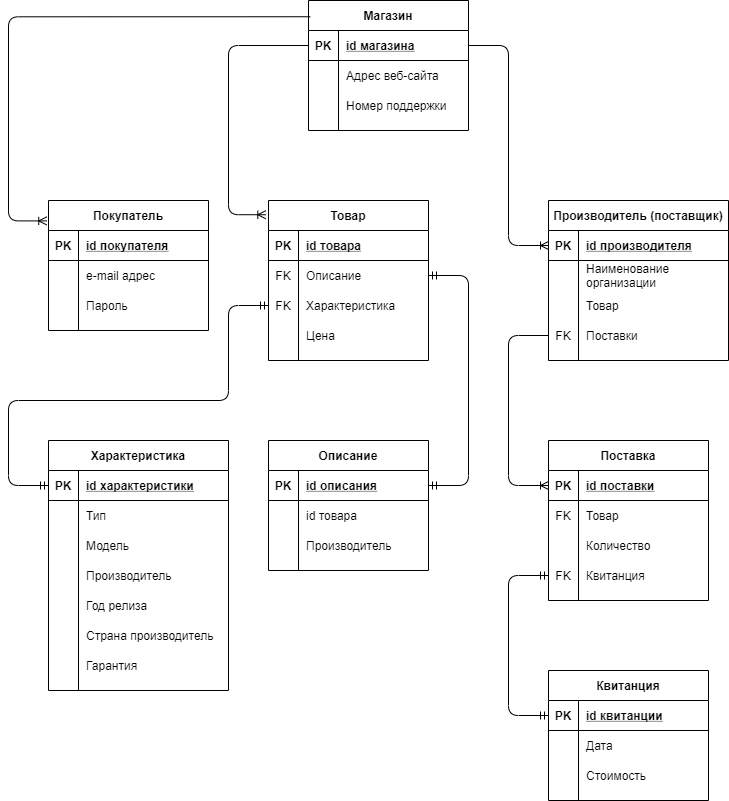


Рисунок 2 — Логическая схема проектируемой БД

## Анализ современных СУБД

Система управления базами данных — совокупность программных и лингвистических средств общего или специального назначения, обеспечивающих управление созданием и использованием баз данных.

СУБД — комплекс программ, позволяющих создать базу данных и манипулировать данными (вставлять, обновлять, удалять и выбирать). Система обеспечивает безопасность, надёжность хранения и целостность данных, а также предоставляет средства для администрирования БД.

Основные функции СУБД

1. управление данными во внешней памяти (на дисках);
2. управление данными в оперативной памяти с использованием дискового кэша;
3. журнализация изменений, резервное копирование и восстановление базы данных после сбоев;
4. поддержка языков БД (язык определения данных, язык манипулирования данными).

Обычно современная СУБД содержит следующие компоненты:

1. ядро, которое отвечает за управление данными во внешней и оперативной памяти и журнализацию;
2. процессор языка базы данных, обеспечивающий оптимизацию запросов на извлечение и изменение данных и создание, как правило, машинно-независимого исполняемого внутреннего кода;
3. подсистему поддержки времени исполнения, которая интерпретирует программы манипуляции данными, создающие пользовательский интерфейс с СУБД;
4. сервисные программы (внешние утилиты), обеспечивающие ряд дополнительных возможностей по обслуживанию информационной системы.

### СУБД PostgreSQL

PostgreSQL — свободная объектно-реляционная система управления базами данных. Существует в реализациях для множества UNIX-подобных платформ, включая AIX, различные BSD-системы, HP-UX, IRIX, Linux, macOS, Solaris/OpenSolaris, Tru64, QNX, а также для Microsoft Windows.

Одной из сильных сторон PostgreSQL является ее архитектура. Как и многие коммерческие СУБД, PostgreSQL может применяться в среде клиент-сервер, что дает массу преимуществ как пользователям, так и разработчикам.

Доступ из приложений к данным базы осуществляется посредством процесса базы данных. Клиентские программы не могут получить доступ к данным самостоятельно, даже если они работают на том же компьютере, на котором выполняется серверный процесс.

Такое разделение клиентов и сервера позволяет построить распределенную систему. Можно отделить клиентов от сервера посредством сети и разрабатывать клиентские приложения в среде, удобной для пользователя. Например, можно реализовать базу данных под UNIX и создать клиентские приложения, которые будут работать в системе Microsoft Windows.

Достоинства

1. Является масштабируемым и способен обрабатывать терабайты данных.
2. Поддерживает формат json.
3. Существует множество предопределенных функций.
4. Доступен ряд интерфейсов.

Недостатки

1. Документация туманна, поэтому, возможно, ответы на некоторые вопросы придется искать в интернете.
2. Конфигурация может смутить неподготовленного пользователя.
3. Скорость работы может падать во время проведения пакетных операций или выполнения запросов чтения.

Идеально подходит для организаций с ограниченным бюджетом.

### СУБД MySQL

MySQL — свободная реляционная система управления базами данных. Эта СУБД позволяет выбирать различные движки для системы хранения, которые позволяют менять функционал инструмента и выполнять обработку данных, хранящихся в различных типах таблиц.

Гибкость СУБД MySQL обеспечивается поддержкой большого количества типов таблиц: пользователи могут выбрать как таблицы типа MyISAM, поддерживающие полнотекстовый поиск, так и таблицы InnoDB, поддерживающие транзакции на уровне отдельных записей. Более того, СУБД MySQL поставляется со специальным типом таблиц EXAMPLE, демонстрирующим принципы создания новых типов таблиц. Благодаря открытой архитектуре и GPL-лицензированию, в СУБД MySQL постоянно появляются новые типы таблиц. Она также имеет простой в использовании интерфейс, и пакетные команды, которые позволяют удобно обрабатывать огромные объемы данных. Система невероятно надежна и не стремится подчинить себе все доступные аппаратные ресурсы.

Достоинства

1. Распространяется бесплатно
2. Прекрасно документирована
3. Предлагает много функций, даже в бесплатной версии
4. Пакет MySQL включен в стандартные репозитории наиболее распространённых дистрибутивов операционной системы Linux, что позволяет устанавливать её элементарно
5. Поддерживает набор пользовательских интерфейсов
6. Может работать с другими базами данных, включая DB2 и Oracle.

Недостатки

1. Отсутствует встроенная поддержка XML или OLAP.
2. Для бесплатной версии доступна только платная поддержка.

Идеально подходит для: организаций, которым требуется надежный инструмент управления базами данных, но бесплатный.

### SQLite

SQLite поддерживает динамическое типизирование данных. Возможные типы значений: INTEGER, REAL, TEXT и BLOB. Также поддерживается специальное значение NULL.

Каждое значение в любом поле любой записи может быть любого из этих типов, независимо от типа, указанного при объявлении полей таблицы. Указанный при объявлении поля тип хранится для справки в его исходном написании, и используется в качестве основы для выбора предпочтений при выполнении неявных преобразований типов на основании похожести этого названия типа на что-либо, знакомое SQLite. В этот алгоритм зашит обширный перечень практикуемых в других СУБД вариантов названий типов данных. Если безопасного преобразования записываемого значения в предпочитаемый тип не получается, SQLite записывает значение в его исходном виде. Для получения значений из базы есть ряд функций для каждого из типов, и если тип хранимого значения не соответствует запрашиваемому, оно тоже, по возможности, преобразуется.

Достоинства

1. Файловая: вся база данных хранится в одном файле, что облегчает перемещение.
2. Стандартизированная: SQLite использует SQL; некоторые функции опущены (RIGHT OUTER JOIN или FOR EACH STATEMENT), однако, есть и некоторые новые.
3. Отлично подходит для разработки и даже тестирования: во время этапа разработки большинству требуется масштабируемое решение. SQLite, со своим богатым набором функций, может предоставить более чем достаточный функционал, при этом будучи достаточно простой для работы с одним файлом и связанной сишной библиотекой.

Недостатки

1. Отсутствие пользовательского управления: продвинутые БД предоставляют пользователям возможность управлять связями в таблицах в соответствии с привилегиями, но у SQLite такой функции нет.
2. Невозможность дополнительной настройки: опять-таки, SQLite нельзя сделать более производительной, поковырявшись в настройках — так уж она устроена.

## Средства разработки

Для создания базы данных была выбрана PyCharm. Для разработки системы был выбран язык Python.

Python — высокоуровневый язык программирования общего назначения, ориентированный на повышение производительности разработчика и читаемости кода. Синтаксис ядра Python минималистичен. В то же время стандартная библиотека включает большой набор полезных функций.

Python поддерживает структурное, обобщенное, объектно-ориентированное, функциональное и аспектно-ориентированно программирование. Основные архитектурные черты — динамическая типизация, автоматическое управление памятью, полная интроспекция, механизм обработки исключений, поддержка многопоточных вычислений, высокоуровневые структуры данных. Поддерживается разбиение программ на модули, которые, в свою очередь, могут объединяться в пакеты.

Python — активно развивающийся язык программирования, новые версии с добавлением/изменением языковых свойств выходят примерно раз в два с половиной года. Язык не подвергался официальной стандартизации, роль стандарта де-факто выполняет CPython, разрабатываемый под контролем автора языка. В настоящий момент Python занимает третье место в рейтинге TIOBE с показателем 10,2 %. Аналитики отмечают, что это самый высокий балл Python за все время его присутствия в рейтинге.