**ҚАЗТҰТЫНУОДАҒЫНЫҢ ЖОҒАРЫ САУДА-ЭКОНОМИКАЛЫҚ КОЛЛЕДЖІ**

**вЫСШИЙ Торгово-экономический колледж казпотребсоюза**

**Logo**

**ОТЧЕТ**

по профессиональной практике

по модулю ПМ 04 «Выполнение практических работ техника– программиста», ПМ 05 «Создание web страниц, сайтов с применением Web технологий»

1304000 «Вычислительная техника и программное обеспечение»

Группа: ВТ 9-20-1

Подготовил(а): Амангельдинов Алиби

Проверил(а): Тлеубаева А.О

Муздубаева Б. Ж.

Оценка:

Нур-Султан 2022

|  |  |  |
| --- | --- | --- |
| **Содержание** | | |
|  | Введение……………………………………………………..……3-4 |  |
|  | Глава 1. Теоретическая часть |  |
|  | 1.1. HTML…………………………………..………..……………5-6 |  |
|  | 1.2 CSS…………………………………………..…………………..6  1.3 JavaScript……………………………………….…..…………6-8  1.4 Python…………………………………………....……………8-9  1.5 Php…………………………………………….………...……..10  1.6 GitHub………………………………………….……...………11  1.7 MySQL………………………………………..……………12-13 |  |
|  |  |  |
|  | Глава 2. Практическая часть |  |
|  | 2.1 ПМ-05………………………………………………..…..14-19 |  |
|  | 2.2 ПМ-04……………………………………………………20-23  2.3 PHP(MySQL) ……………………………………….………24 |  |
|  | Заключение…………………………………………………..……25 |  |
|  | Список литературы………………………………………….……26 |  |
|  |  |  |

**Введение**

Цели практики: изучить теорию построения HTML документов с использованием стилей CSS и скриптов JS, а так-же особенностей и возможностей языка Python, и построения СУБД на языках php и mysql, и отработать практические задания по этим темам.

Компьютеры так прочно вошли в нашу жизнь, что мы уже не можем представить ее без них. Трудно поверить, что только полвека назад появились первые компьютеры, которые были медлительными и громоздкими. Сейчас у нас есть огромное многообразие - от ультратонких ноутбуков до мощных игровых компьютеров.

Сейчас прогресс ушел так далеко, что техника может интуитивно понимать запрос пользователя. Однако стоит помнить, что все программы, браузеры и сайты имеют в основе языки программирования.

В наши дни существует около 8 тысяч таких языков, многие из которых понятны лишь авторам. Однако есть такие языки программирования, которые стали международными и используются миллионами людей ежедневно.

Без языка программирования ни один компьютер работать не будет. С его помощью создаются определенные математические алгоритмы, которые помогают компьютеру выполнять команды пользователя.

* Язык программирования выглядит совсем не так, как мы привыкли представлять себе язык - это набор различных символов, которые преобразуются в понятный компьютеру код.
* Классификация языков программирования от низких до высоких уровней зависит от того, настолько близок язык к человеку. Чем меньше язык программирования понятен человеку, тем по уровню он ниже.
* Большая часть языков преобразуется в понятный для машины код с помощью программ-трансляторов. С их помощью прописываются лексические, семантические и синтаксические правила, которые определят, какие действия запросит пользователь и внешний вид программы.
* Язык программирования - неотъемлемая часть работы любого компьютера. В современном мире с ними работают программисты, которые прописывают коды и создают программы с использованием сложных кодовых конструкций.

Для обычного человека написание даже первой странички сайта покажется странной комбинацией различных символов. Для программиста этот код позволяет задать компьютеру нужную команду и выполнить ее. На языке программирования обычный пользователь общается с компьютером.

Первыми "ласточками" в истории языка программирования считаются такие обычные вещи, как музыкальная шкатулка или жаккардовый ткацкий станок. Они появились еще в XVIII-XIX веках.

Начало же современному программированию было дано в 1930-1940-е годы, когда были написаны лямбда-исчисление и машина Тьюринга. То время можно назвать началом истории разработки языков программирования, ведь компьютеры становились все популярнее. В 1940-е годы появились первые электрические компьютеры, и тогда же немецкий инженер Ц. Кузе разработал Планкалькюль, первый высокоуровневый язык.

В 1950-е годы были разработаны машинные коды, которые считаются языками программирования первого поколения. Но их приходилось переписывать для каждой ЭВМ отдельно, так что первым реально работающим языком программирования можно считать «Краткий код». Он первым стал представлять собой не набор математических кодов, а выражения, которые потом перерабатывались в код.

HTML (HyperText Markup Language) - язык разметки гипертекста - предназначен для создания Web-страниц. Под гипертекстом в этом случае понимается текст, связанный с другими текстами указателями-ссылками.

HTML представляет собой достаточно простой набор кодов, которые описывают структуру документа. HTML позволяет выделить в тексте отдельные логические части (заголовки, абзацы, списки и т.д.), поместить на Web-страницу подготовленную фотографию или картинку, организовать на странице ссылки для связи с другими документами.

HTML не задает конкретные и точные атрибуты форматирования документа. Конкретный вид документа окончательно определяет только программа-браузер на компьютере пользователя Интернета.

HTML также не является языком программирования, но web-страницы могут включать в себя встроенные программы-скрипты на языкахJavascript и Visual Basic Script и программы-апплеты на языке Java.

Даже, если вы не собираетесь в дальнейшем редактировать "вручную" текст HTML (предполагая использовать графические редакторы), знание языка HTML даст вам возможность как лучше использовать эти средства, так и увеличит ваши шансы сделать HTML-документ более доступным и "читаемым" при просмотре браузерами разных фирм.

Основными компонентами HTML являются:

Тег (tag). Тег HTML это компонент, который командует Web- броузеру выполнить определенную задачу типа создания абзаца или вставки изображения.

Атрибут (или аргумент). Атрибут HTML изменяет тег. Например, можно выровнять абзац или изображение внутри тега.

Значение. Значения присваиваются атрибутам и определяют вносимые изменения. Например, если для тега используется атрибут выравнивания, то можно указать значение этого атрибута. Значения могут быть текстовыми, типа left или right, а также числовыми, как например ширина и высота изображения, где значения определяют размер изображения в пикселях.

Теги представляют собой зарезервированные последовательности символов, начинающиеся с < (знака меньше) и заканчивающиеся > (знаком больше). Закрытие тега отличается от открытия только наличием символа '/'.

Предположим, у нас есть гипотетический атрибут форматирования текста, управляемый кодом <X>, и мы хотим применить его к словам "Это мой текст". HTML-последовательность кодов и собственно текста будет выглядеть так: <X>Это мой текст</X>. Теги могут вкладываться друг в друга иерархически, но без пересечений, то есть допустимо вложение вида <teg1><teg2></teg2> </teg1>, но не<teg1><teg2> </teg1></teg2>.

Действие вложенных тегов объединяется. Например, если внутрь тега, создающего жирное начертание шрифта, вложен тег курсива, то в результате получится жирный курсив.

CSS (Cascading Style Sheets, каскадные таблицы стилей) — язык описания внешнего вида HTML-документа. Это одна из базовых технологий в современном интернете. Практически ни один сайт не обходится без CSS, поэтому HTML и CSS действуют в единой связке. Каскадные таблицы стилей работают с HTML, но это совершенно другой язык. HTML структурирует документ и упорядочивает информацию, а CSS взаимодействует с браузером, чтобы придать документу оформление.

На языке HTML мы создаем размеченный текст — документ с гиперссылками, таблицами, маркированными списками, разными начертаниями шрифтов, заголовками, подзаголовками и так далее. Получаем «простыню» текста с таблицами и иллюстрациями. Интернет изобрели ученые, и для них такое положение вещей было приемлемым. Но всё изменилось, когда WWW пошел в массы и свои странички начали создавать простые пользователи, которые хотели индивидуальности и самовыражения, а также коммерческие компании со своими корпоративными стандартами оформления. В общем, веб-страницам понадобилось индивидуальное оформление: стиль.

В таблице стилей мы прописываем стили (правила) оформления. Каждое правило состоит из селектора и блока объявлений. Селектор определяет, на какие части документа распространяется правило. Блок объявлений помещается в фигурные скобки и состоит из одного или более объявлений, разделенных точкой с запятой. Нет одной общепринятой методологии CSS. Их несколько, а со временем возможно появление новых.

JavaScript – это мультипарадигменный язык программирования, который обычно применяется в качестве встраиваемого инструмента для программного доступа к различным объектам приложений.

С точки зрения веб-разработки, без знаний этой технологии невозможно заниматься созданием современных интерактивных сайтов. Язык JS – это то, что «оживляет» разметку страниц (HTML) и пользовательский функционал (CMS) сайтов. С помощью этого языка реализуется возможность реакции страницы или отдельных ее элементов на действия посетителя. Сегодня JavaScript является базовым языком программирования для браузеров. Он полностью совместим с операционными системами Windows, Linux, Mac OS, а также всеми популярными мобильными платформами.

Стандартный алгоритм работы выглядит следующим образом:

1. Пользователь выполнил определенное действие.
2. Браузер определил событие.
3. Активируется JS-код.
4. На странице происходит заданное изменение.

Задача программиста заключается в том, чтобы создать обработчики для всех событий, на которые должен реагировать сайт при взаимодействии с пользователем. Для наглядности приведем два примера типичных сценариев.

1. Пользователь нажимает левую кнопку мыши.
2. Браузер фиксирует событие onclick.
3. Активируется функция changePhoto.
4. В окне просмотра фотографий изменяется изображение.

Если обработчик не внедрен в код, то алгоритм работы будет примерно следующим:

1. Пользователь совершает клик.
2. Браузер фиксирует событие onkeydown.
3. Для его обработки нет специального JS-кода.
4. После клика ничего не происходит.

Нужно отметить, что программирование не всегда привязывается к активным действиям пользователя.

К примеру, JavaScript-код может срабатывать при полной загрузке страницы или после определенного времени нахождения на сайте. Эти возможности активно используются для создания всплывающих Pop-up элементов и чатов.

[Python](https://skillbox.ru/media/code/kratkaya-istoriya-python/) — это скриптовый язык программирования. Он универсален, поэтому подходит для решения разнообразных задач и многих платформ, начиная с iOS и Android и заканчивая серверными ОС.

Это интерпретируемый язык — он не компилируется, то есть до запуска представляет из себя обычный текстовый файл. Программировать можно практически на всех платформах, язык хорошо спроектирован и логичен. Разработка идёт в разы быстрее, потому что кода здесь куда меньше, чем на других языках. И ещё Python отлично подходит новичкам.

Его можно встретить в вебе и на мобильных устройствах, в приложениях и решениях, связанных с машинным обучением (нейросети и искусственный интеллект), а также в качестве встроенной системы.

Чаще всего Python используется в веб-разработке. Для работы с ним подключают фреймворки: Pyramid, Pylons, TurboGears, Flask, CherryPy и — самый популярный — Django.

Существуют и движки для создания сайтов на Python:

* Abilian SBE;
* Ella;
* Saleor;
* Wagtail;
* Django-CMS.

Также на Python пишут парсеры для сбора информации в интернете. Хоть язык не компилируется, с помощью него создают десктопные программы. Вот, к примеру, что было разработано на Python:

* GIMP — визуальный редактор на Linux;
* Ubuntu Software Center — центр приложений в ОС Ubuntu (один из дистрибутивов Linux);
* BitTorrent до 6 версии — менеджер торрент-закачек (позже программу переписали на C++, но сети peer-to-peer всё ещё работают на Python);
* Blender — программа для создания 3D-графики.

Мобильные приложения

Мобильная разработка на Python менее популярна. Для Android чаще пишут на Java, C#, C++ или Kotlin, а для iOS — на Swift или Objective-C. На Python обычно программируют серверную часть приложения. Например, клиент Instagram для iOS написан на Objective-C, а сервер — на Python.

Многие компьютерные игры были полностью или частично написаны на Python. Существует заблуждение, что этот язык не подходит для серьёзных проектов, но на самом деле он использовался в разработке таких хитов, как:

* Battlefield 2;
* World of Tanks;
* Civilization IV;
* EVE Online.

Несмотря на возможность реализации пользовательского интерфейса и работы с графикой, на Python в основном пишут скрипты — например, взаимодействия персонажей, запуска сцен, а также обработки событий.

На Python разрабатывают встроенные системы для различных устройств. Например, язык прижился в Raspberry Pi (компьютер размером с карту памяти) и в «Сбербанке» для управления банкоматами.

Еще проекты со встроенной системой на Python:

* The Owl Embedded Python System;
* Python Embedded Tools;
* Embedded Python.

Язык применяется во встроенных системах станков с ЧПУ, средствах автоматического регулирования (температуры, расхода жидкостей, давления и так далее) и в телекоммуникационном оборудовании.

PHP: Hypertext Preprocessor (изначально Personal Home Page Tools, «Инструменты для создания персональных веб-страниц») — один из лидирующих языков современной веб-разработки. Его отцом считается датский программист Расмус Лердорф, который в 1994 году создал набор скриптов на Perl — ту самую «персональную домашнюю страницу», которая легла в основу PHP.

Со временем к работе над ним подключились разработчики со всего мира. PHP — один из старейших языков в рамках open source-проекта. Сейчас его поддерживает и разрабатывает группа энтузиастов во главе с компанией [Zend Technologies](http://www.zend.com/" \t "_blank). Ей руководят Зеев Сураски и Энди Гутманс: в 1997 году они создали третью версию PHP и активно развивают язык по сегодняшний день. Первое свойство: интерпретируемость

Главная характеристика PHP - интерпретируемость. В отличие от Java, которая компилируется, а затем запускается в работу, PHP создается во время обращения к нему. Человек открывает сайт, на сервер посылается запрос, и в это время компилируется код. Каждый скрипт компилируется в реальном времени, а затем выполняется.

Благодаря этому свойству PHP — очень гибкий язык. Например, Java-программист вводит название класса один раз — теперь оно неизменно. PHP же компилируется в рантайме и можно подставить любое название класса из переменной.

Однако ради гибкости разработчик жертвует скоростью работы сайта. В случае с Java программу создают, а затем запускают. В случае с PHP приходится занимать время пользователя на процесс компиляции кода.

Второе свойство: динамическая типизация

Языки программирования бывают:

 1) со статической типизацией;

 2) с динамической типизацией.

В первом случае тип переменной определен жестко и заранее. Если это число, программист напишет тип int. Если строка — string. Во втором случае переменная может быть одновременно числом, строкой, массивом, объектом — чем угодно. Одной переменной можно присвоить число, затем массив, объект — и язык программирования это позволит. PHP относится ко второй группе.

GitHub - это система управления проектами и версиями кода, а также платформа социальных сетей, созданная для разработчиков. Но для чего используется GitHub? Ну, среди прочего, это позволяет вам работать совместно с другими людьми по всему миру, планировать свои проекты и отслеживать свою работу. GitHub также является одним из крупнейших онлайн-хранилищ совместной работы по всему миру.

Когда разработчики создают новый проект, они всегда продолжают вносить обновления в код. Даже после запуска проектов им всё равно нужно обновлять версии, исправлять ошибки, добавлять новые функции и т. д. Система контроля версий помогает отслеживать изменения, внесённые в базу кода. Более того, он записывает, кто внёс изменения и может восстановить стёртый или изменённый код.

Если Git - это сердце GitHub, то Hub - это его душа. Концентратор в GitHub - это то, что превращает командную строку, такую ​​как Git, в крупнейшую социальную сеть для разработчиков. Помимо участия в определённом проекте, GitHub позволяет пользователям общаться с единомышленниками. Вы можете следить за людьми и смотреть, что они делают или с кем они общаются.

Репозиторий или хранилище - это каталог, в котором хранятся файлы вашего проекта. Он может быть расположен в хранилище GitHub или в локальном хранилище на вашем компьютере. Вы можете хранить файлы кодов, изображения, аудио или всё, что связано с проектом, в хранилище.

Ветка - это копия вашего репозитория. Вы можете использовать ветку, когда хотите сделать разработку изолированно. Работа с веткой не повлияет на центральное хранилище или другие ветки. Если вы сделали эту работу, вы можете объединить свою ветку с другими ветками и центральным репозиторием, используя запрос на извлечение.

Запрос на извлечение означает, что вы сообщаете другим, что вы передали изменения, внесённые вами в ветке, в главный репозиторий. Соавторы хранилища могут принять или отклонить запрос на извлечение. После его открытия вы можете обсудить и проанализировать свою работу с соавторами.

Форкинг репозитория означает, что вы создаёте новый проект на основе существующего репозитория.

MySQL — это реляционная база данных (СУБД). Реляционная — значит внутри неё есть данные, которые связаны между собой, и эту связь можно представить в виде таблиц. Технически MySQL — это много таблиц, как-то связанных между собой. Например, одна отвечает за товары, другая — за покупки, третья — за клиентов.

Знать MySQL нужно всем, кто занимается разработкой веб-приложений и сайтов. Это очень распространённая технология — если ваше приложение или сайт имеет в каком-либо виде личный кабинет или просто хранит любые данные, то почти наверняка в нём будет использоваться MySQL. Вы можете обойтись и без конкретно этой системы управления. Можно использовать PostgreSQL или NoSQL. Можно просто хранить данные у клиента или в «сыром» файлике. Но если вы хотите делать систему, которую будет легко поддерживать и передать другим людям для доработки и развития, — скорее всего, вы выберете MySQL.

Связи между таблицами показывают, как одни данные могут зависеть от других. В зависимости от того, как вы настроите связи, база может получать разные результаты и искать данные. В связи может участвовать одна запись, а может сразу много.

Один к одному. Это самый простой вид связи, который говорит: одной записи из этой таблицы соответствует только одна запись из другой таблицы. Если мы сделаем новую таблицу с фотографиями клиентов, то каждой фотографии будет соответствовать только один клиент и наоборот.

Один ко многим. Когда у нас есть таблица с клиентами и таблица с их покупками, тут работает связь «один ко многим». Это значит, что у нас одной записи о клиенте соответствует несколько записей об их покупках, например, если он их сделал в разное время. Благодаря этой связи мы можем вывести все покупки для каждого клиента в отдельности.

Ещё пример — художники и картины. Каждая картина принадлежит только одному художнику, но одному художнику может принадлежать много разных картин.

Многие ко многим. А вот это хитрая связь — для неё нужна отдельная таблица. Смысл такой: мы делаем промежуточную таблицу, в которой соединяем данные из одной таблицы с данными из другой. Получается, что в новой таблице как бы нет данных — в ней только связи.

Допустим, вы ведёте свой список дел в ежедневнике, где можно ставить метки для дел. Метки помогают понять, что за дело перед вами, и выглядят примерно так: «в дороге», «позвонить», «на неделе», «подписать у Иваныча» и «за компьютером». Их можно назначить любой задаче — одну метку, две или все сразу. Получается так:

* одна метка может стоять на множестве разных задач
* у одной задачи может быть много разных меток.

Это значит, что мы связали множество меток со множеством задач и теперь можем искать одно через другое.

В MySQL может храниться что угодно, если вы можете настроить связи между данными. Вы можете хранить в такой базе заметки, фото, музыку, списки дел, задачи на год и все лекции по теории вероятности. Весь вопрос в том, чтобы вы понимали, как вы будете это использовать дальше.

Например, все наши статьи в «Коде» хранятся в MySQL-базе, с которой мы работаем [через Вордпресс](https://thecode.media/what-is-wordpress/). Там же есть информация и об авторах, и о картинках для статей, о дате публикации и о многом другом. Чтобы вы прочитали эту статью, сайт обратился к базе данных, взял оттуда статью, правильно её обработал и показал вам.

Другие используют MySQL для работы с клиентской базой — в бизнесе, поликлиниках или системах учёта товаров. Самой базе всё равно, что в ней хранится и как вы этим пользуетесь. База данных — это просто способ связать данные вместе, а потом найти в них то, что нужно.

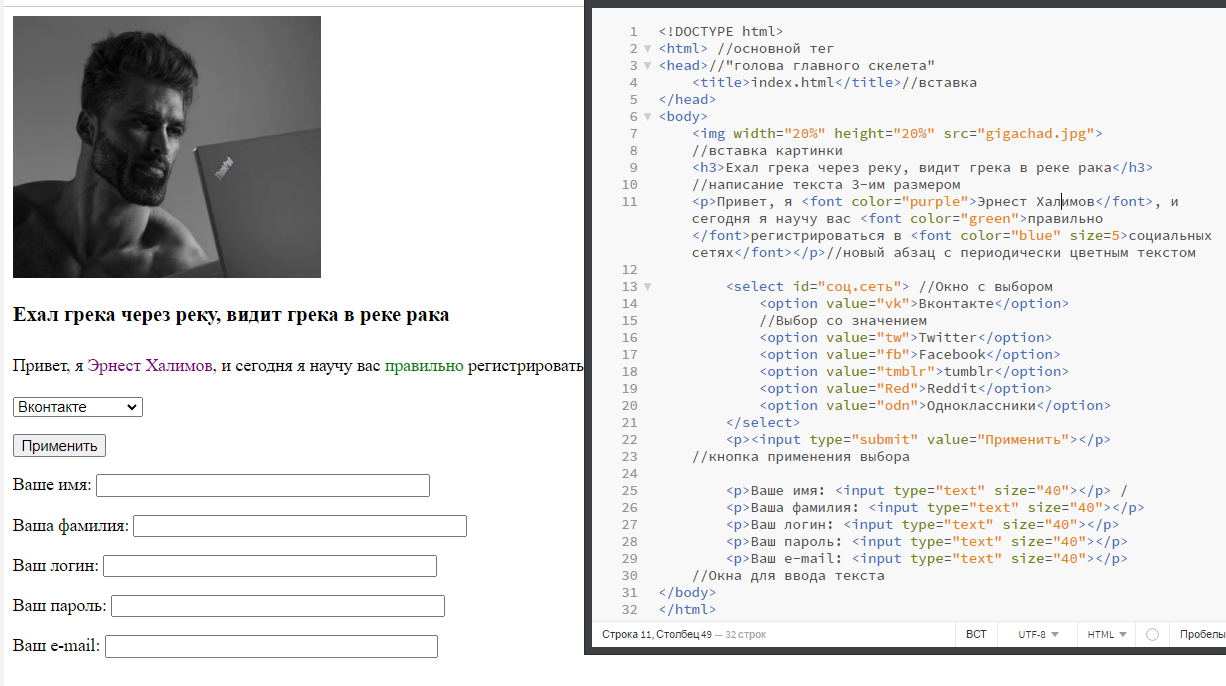
С момента своего появления в 1995 году, MySQL была бесплатной, простой и предсказуемой системой управления базами данных. Это привело к тому, что её использовали много компаний по всему миру, что сделало её негласным стандартом для баз данных.

Ещё в MySQL встроены системы безопасности и разграничения доступа. Например, можно сделать так, чтобы менеджер мог только вносить данные, руководитель отдела — изменять их, но не удалять, а директор мог делать с данными что угодно.

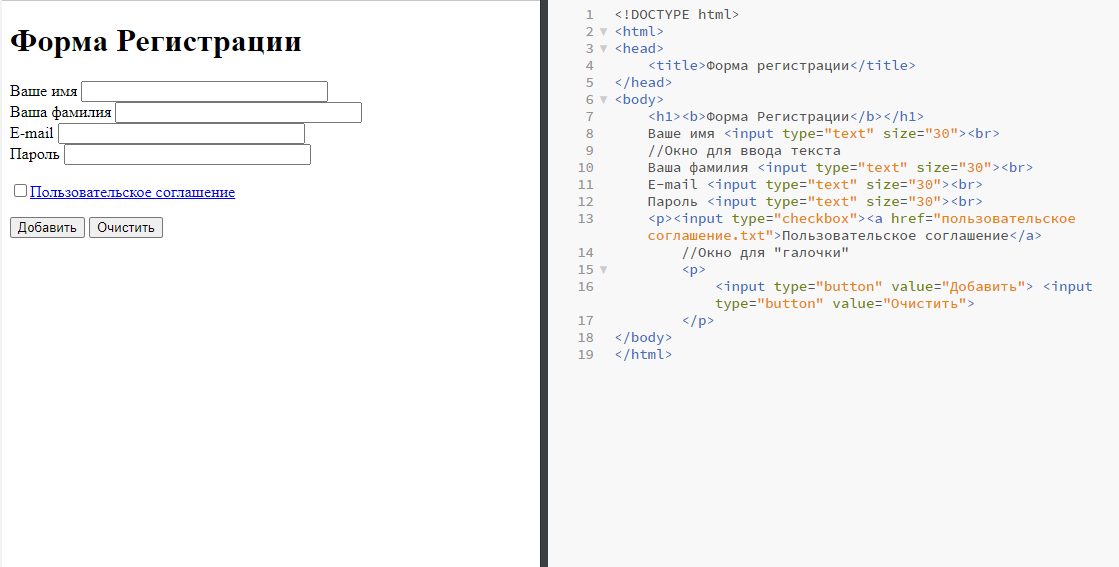
Но основная причина популярности MySQL — полная поддержка SQL-языка.

Практическая работа 1. HTML и CSS

Задание 1) Создание HTML-документа (рис.1)

 (Рис. 1 – решение задания 1)

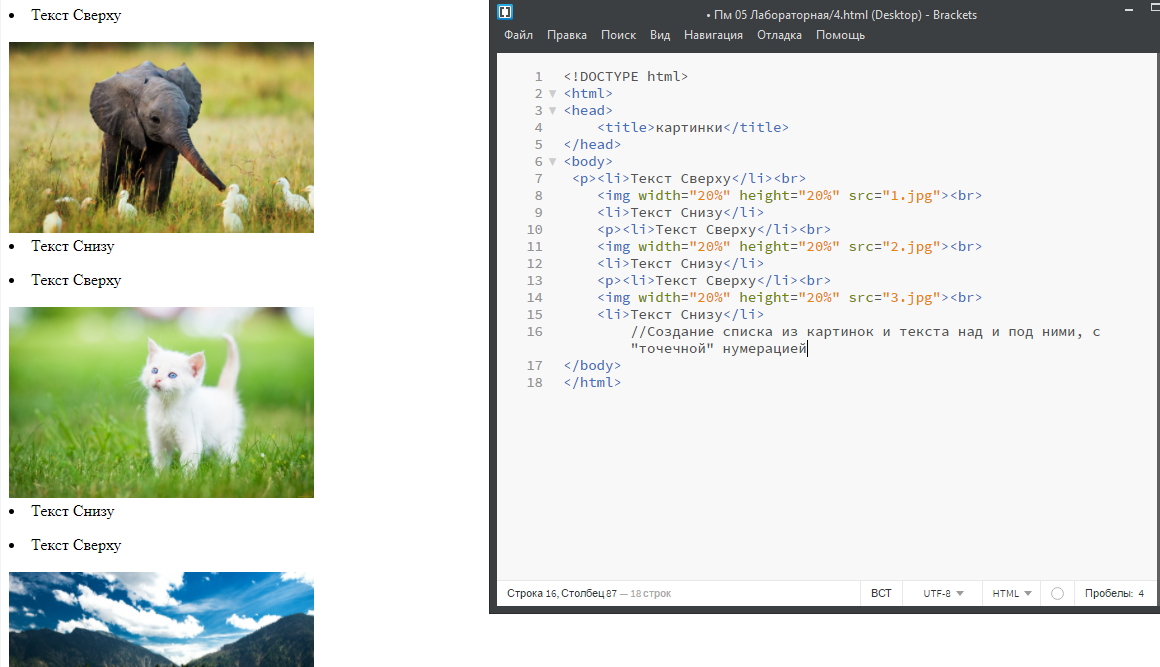
Задание 2) Форма регистрации (рис.2)

 (Рис.2 - решение задания 2)

Задание 3) Разметка HTML-страницы (рис.3)

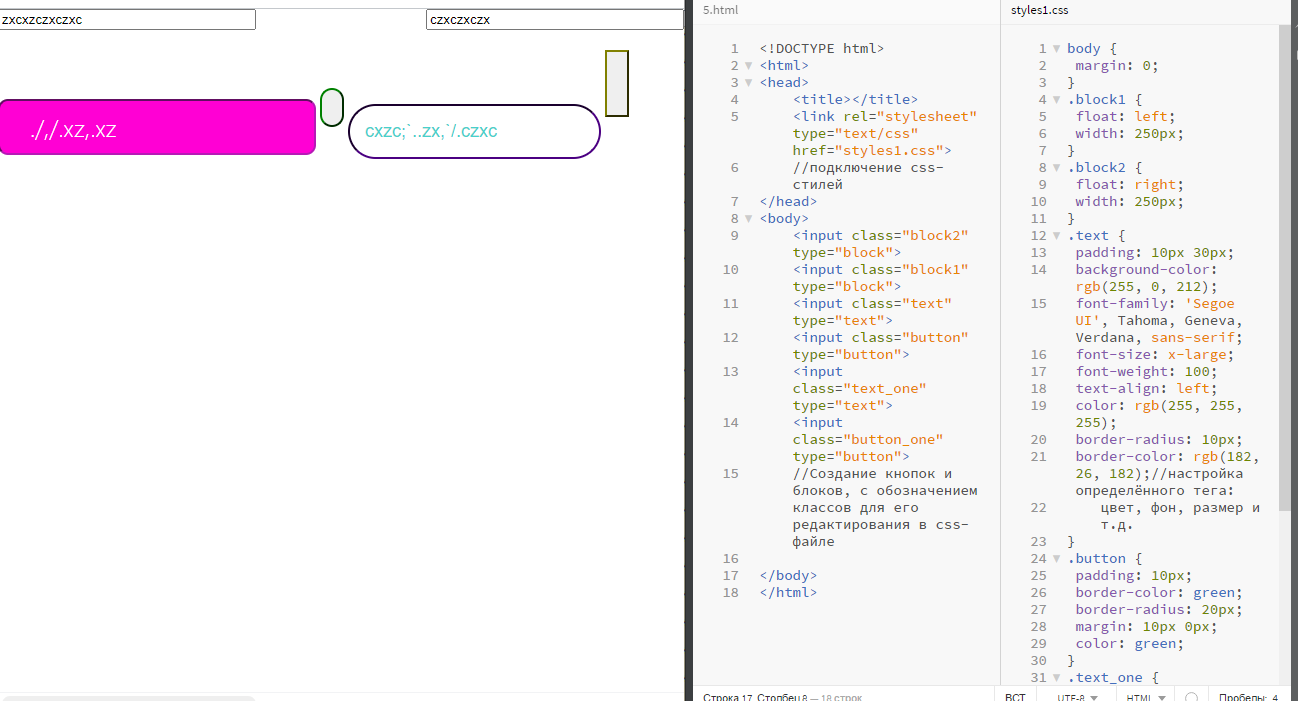
 (Рис.3 – решение задания 3)

Задание 4) Список изображений (рис.4)



(Рис.4 – решение задания 4)

Задание 5) Создание блоков (рис.5)



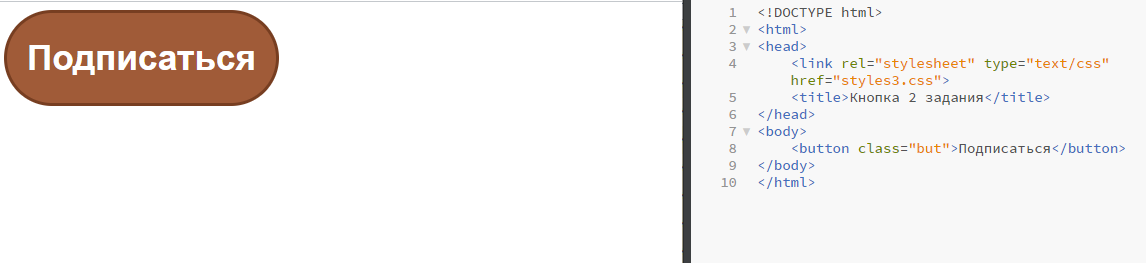
(Рис.5 – решение задания 5)

Задание 6) Красивая кнопка (рис.6)



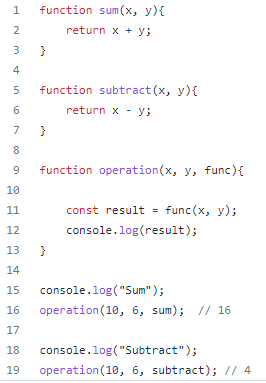
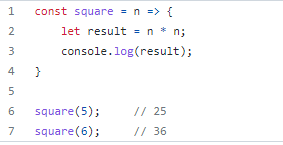
(Рис.6 – решение задания 6)

Задание 7) Фиксированная кнопка (рис.7)

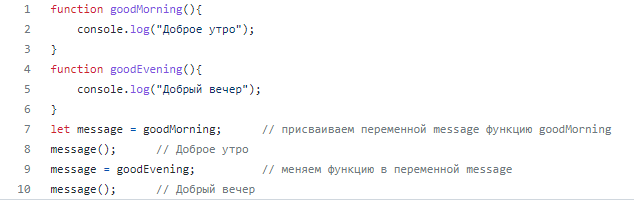


(Рис.7 – решение задания 7)

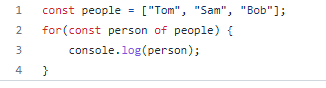
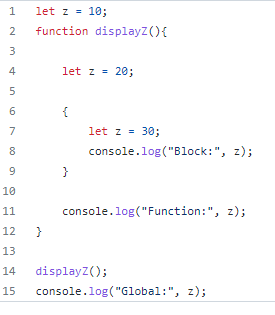
Практическая работа 2 – JavaScript

(Рис.1 – стрелочные функции ) ( Рис.2 – параметры функций)



(Рис.3 – функциональное программирование функции)

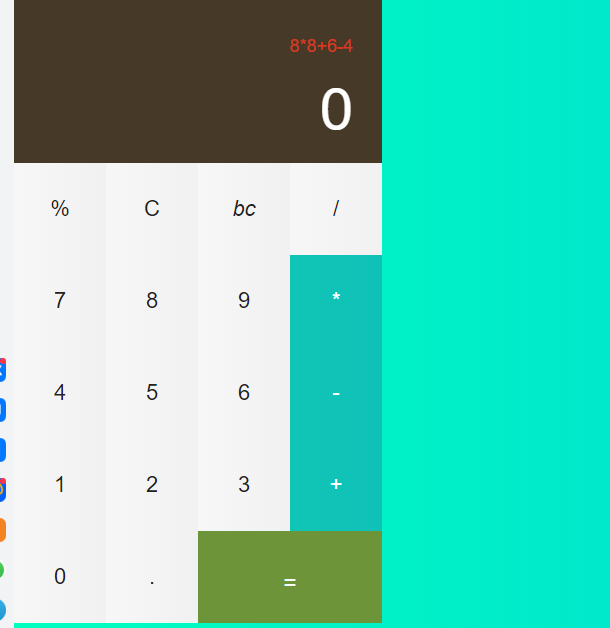
 

(Рис. 4 – циклы) (Рис.5 – область видимых переменных)

Практическая работа HTML 3 - Калькулятор



(Рис.1 – JS, скриптовая часть) ( Рис.2 – HTML и CSS часть)

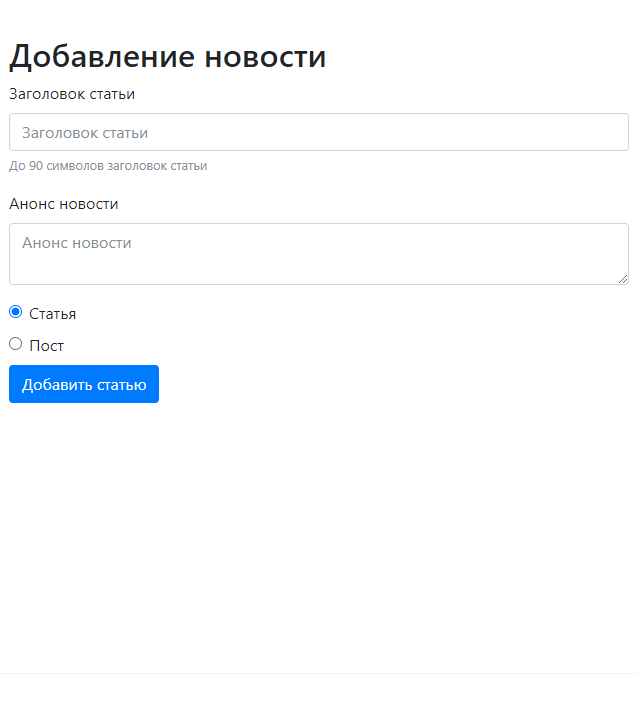


(Рис.3 – рабочий калькулятор)

Практическая работа HTML 4 – Новостная форма

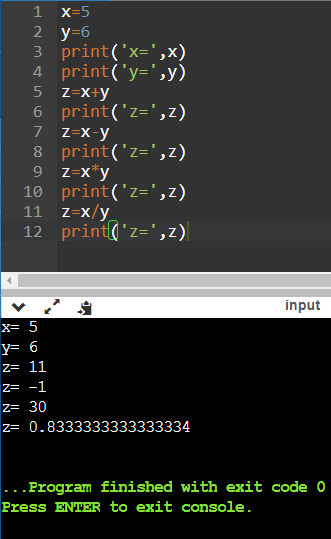
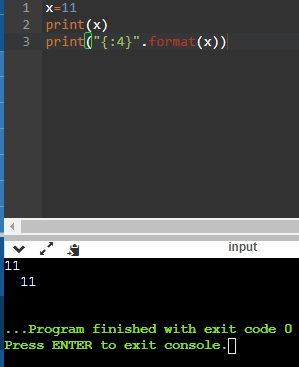


(Рис.1 – HTML и CSS часть)

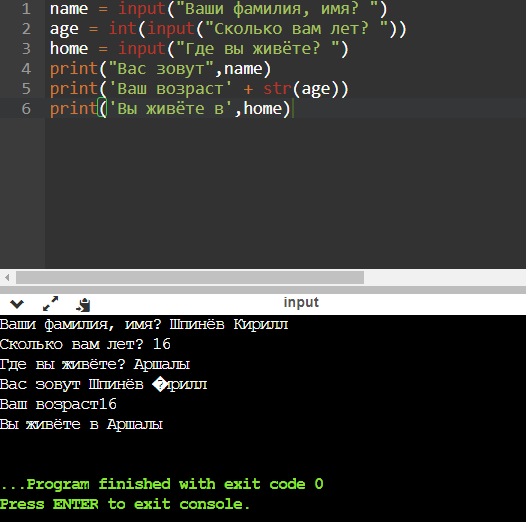
 

(Рис.2 – JS часть) (Рис.3 – рабочий сайт)

Лабораторная работа 1. Python

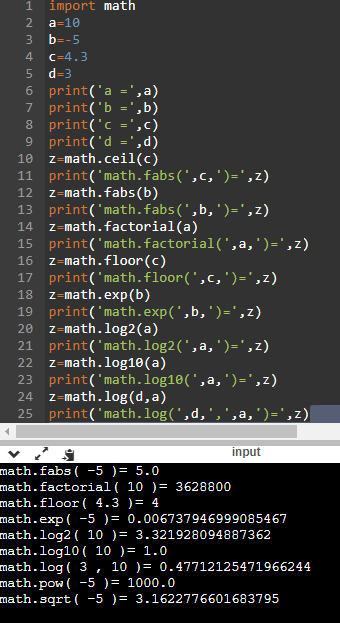
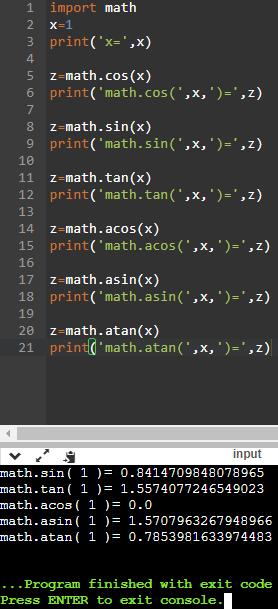
 

(Рис.1 – пример задачи) (Рис.2 – вывод через format)

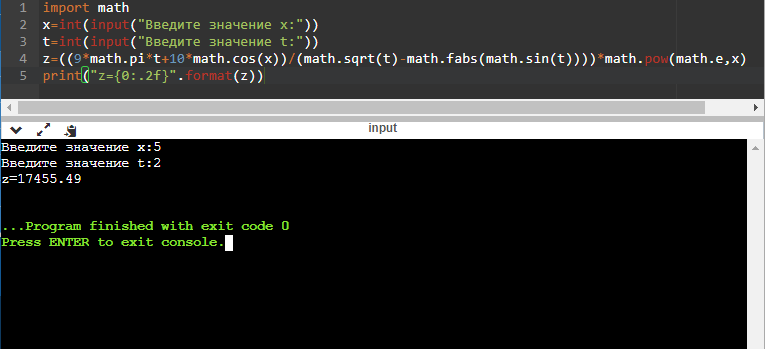


(Рис.3 – решение задачи по варианту)

Лабораторная работа 2. Python

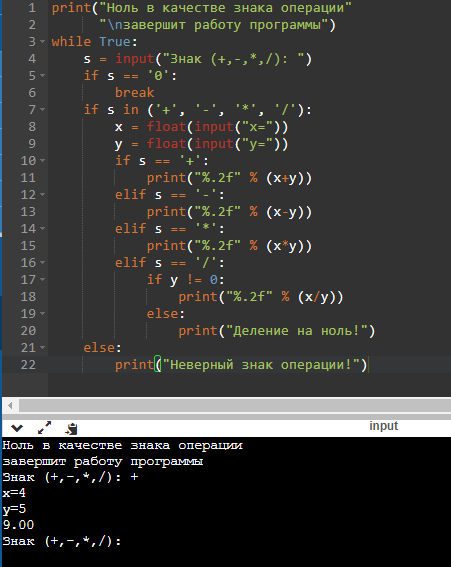
 

(Рис.1 – пример программы) (Рис.2 – 2-й пример программы)



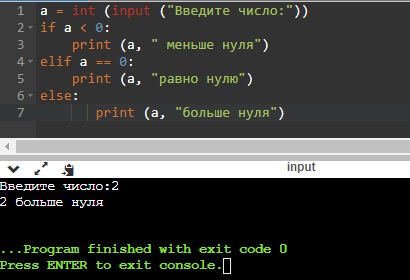
(Рис.3 – решение выражения с подставлением переменных)

Лабораторная работа 3. Python



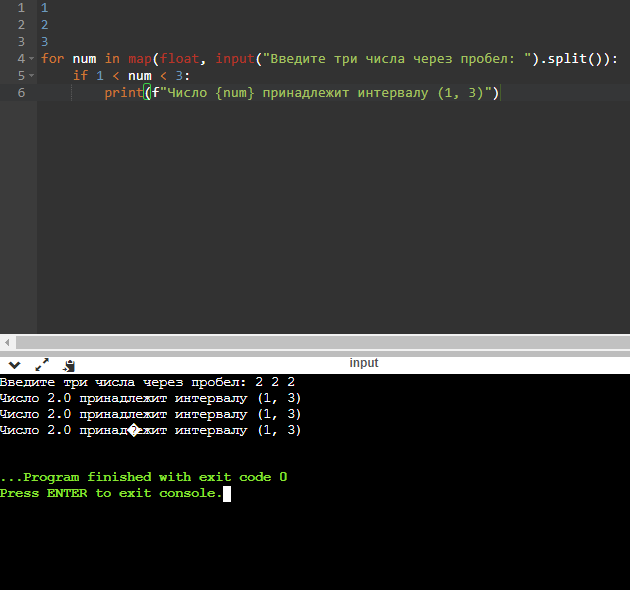
(Рис.1 – мини-калькулятор)

Лабораторная работа 4. Python



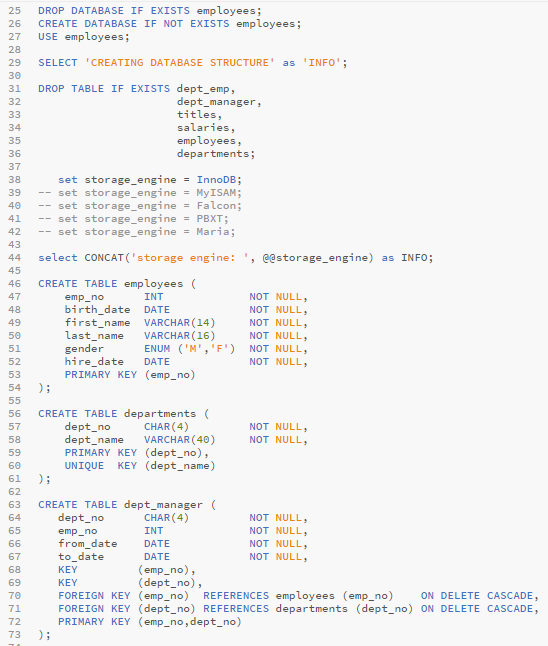
(Рис.2 – пример if else)

Самостоятельная работа по 3 лабораторной. Python

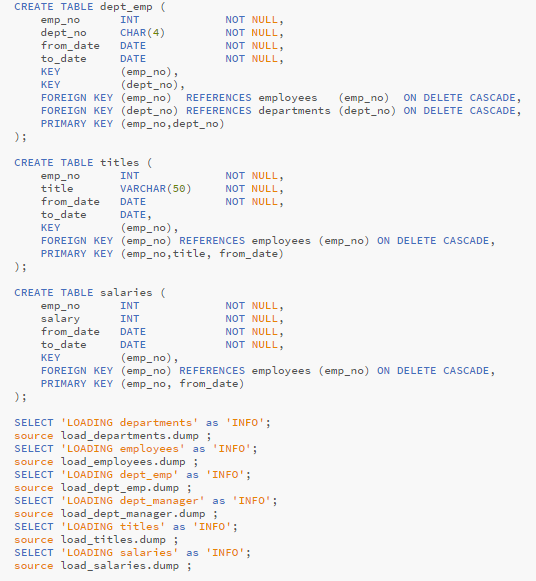


(Рис.1 – условные операторы)

Практическая работа 1. Php (Mysql)



(Рис.1 – создание базы данных сотрудников 1)



(Рис. 2 – создание базы данных сотрудников 2)

**Заключение**

Данная практика помогла мне освоить применение CSS стилей и JS скриптов в HTML-документах, например: красивое оформление сайта, создание новостной формы и рабочий калькулятор. Язык Python тоже оказался полезным, ведь с помощью него можно создавать игры и сайты, на нём я смог создать мини-калькулятор и подставлять значения переменных, а при помощи языка Mysql я создал базу данных.

**Список литературы**

<https://thecode.media/mysql/>

<https://gb.ru/blog/baza-dannykh-mysql/>

<https://gb.ru/posts/kak-ustroen-i-rabotaet-github>

<https://skillbox.ru/media/code/dlya_chego_nuzhen_python/>

<https://htmlweb.ru/java/js.php>