

**Автономная некоммерческая организация  
дополнительного профессионального образования  
«АКАДЕМИЯ ДОПОЛНИТЕЛЬНОГО ПРОФЕССИОНАЛЬНОГО ОБРАЗОВАНИЯ»**

**ПРИНЯТО**  
на заседании Педагогического совета  
(протокол № 2 от 4 октября 2022г.)

**УТВЕРЖДЕНО**  
приказом ректора от 6.10.2022г. № 36/2

**Дополнительная общеразвивающая программа  
«Классическое программирование на языке Python»**

Срок освоения: 1 год  
Возраст обучающихся: 14-18 лет

Разработчик программы: Маринина Н.Ю.,  
педагог дополнительного образования

Внутренняя экспертиза проведена. Программа рекомендована к рассмотрению на Педагогическом совете организации.

Петрова О.Е., директор Учебного центра вычислительной техники

Кочнева И.В., заместитель директора по УМР

«3» октября 2022г.

## 1. Пояснительная записка

### 1.1. Основные характеристики дополнительной общеразвивающей программы

Дополнительная общеразвивающая программа «Классическое программирование на языке Python» (далее – программа) имеет *техническую направленность* и ориентирована на развитие интереса детей к инженерно-техническим и информационным технологиям, повышению уровня технической грамотности в области инженерных профессий.

*Адресат программы:* школьники в возрасте 14-18 лет, желающие освоить синтаксические основы языка Python и общие принципы создания Windows-приложений. Программа составлена с учетом возрастных особенностей учащихся, исходный уровень владения компьютером значения не имеет. Прием на обучение осуществляется по желанию учащегося и/или его законного представителя.

*Актуальность программы* в том, что в соответствии с государственной программой «Развития образования» тематика программы отвечает потребностям современного общества. Алгоритмический подход при решении задач — это залог успешного применения средств вычислительной техники в различных областях человеческой деятельности и, в первую очередь, при разработке программного обеспечения. Умение разрабатывать алгоритмы — это основа процесса программирования, фундамент обучающегося в будущей успешной деятельности в качестве разработчика программного обеспечения. С другой стороны, умение алгоритмически мыслить принесет пользу школьнику не только в сфере узконаправленной деятельности программиста, но и будет способствовать общему интеллектуальному развитию, т. к. учит структурировать информацию, дисциплинирует мышление, способствует развитию памяти, обостряет внимание.

Программа призвана содействовать профессиональному самоопределению учащихся, реализации индивидуального потенциала, достижению сбалансированности между профессиональными интересами учащегося, его психофизическими особенностями и потребностями рынка труда.

*Отличительная особенность программы* заключается в ее практической направленности и приближенности к реальным запросам экономики будущего, которой требуются качественно подготовленные специалисты в области прототипирования, конструирования и 3D-моделирования. Программа направлена на подготовку глубоко профориентированных абитуриентов, которые смогут сразу после диплома выполнять прикладные профессиональные задачи.

Программа носит профориентационный характер и построена с учетом отсутствия требований к первоначальному уровню знаний учащихся.

*Уровень освоения программы:* общекультурный. Программа направлена на формирование общей культуры учащихся в области информационных технологий и удовлетворяет индивидуальные потребности в интеллектуальном развитии.

*Объем и срок освоения программы:* 108 учебных часов, 1 год.

*Цели программы* – формирование навыков алгоритмического и творческого мышления, целостного представления о глобальном информационном пространстве и принципах получения информации, конструкторских и исследовательских навыков активного творчества с использованием современных технологий.

Достижение поставленных целей осуществляется путем решения следующих *задач*:  
*обучающие:*

формирование базовых теоретических понятий и знаний основ написания программ;

знакомство с этапами и основными подходами разработки программного обеспечения, с технологиями программирования и классическими алгоритмами для различных структур данных;

обучение правилам использования стандартных элементов интерфейса программных продуктов, работающих под Windows, синтаксическим конструкциям различных языков программирования, научить приемам ведения проектной деятельности при разработке программного обеспечения.

– *развивающие:*

- развивать навыки «мозгового штурма», устного счета, творческого инициативного мышления и способности систематизировать информацию;
- развивать логическое и пространственное мышление;

– *воспитательные:*

- воспитывать силу воли и настойчивости при выполнении поставленных задач;
- обучать работе в команде, стрессоустойчивости при организации защиты проекта;
- развивать инициативность, самостоятельность, системный подход в решении задачи;
- способствовать самоопределению учащегося в профессиональной сфере.

По завершении обучения учащийся должен достигнуть следующие *результаты:*

– *метапредметные:*

проявление интереса и положительную мотивацию к изучению информационных технологий;

самостоятельная работа с источниками информации, обобщение и систематизация полученной информации, интегрирование ее в личный опыт;

совершенствование навыков работы с дополнительной литературой, использования различных Интернет-ресурсов;

выдвижение версий решения проблемы, осознание конечного результата, выбор средства достижения цели из предложенных или их самостоятельный поиск;

составление (индивидуально или в группе) плана решения проблемы;

самостоятельно осознание причин своего успеха или неуспеха и находить способы выхода из ситуации неуспеха;

умение оценить степень успешности своей индивидуальной образовательной деятельности;

самостоятельная организация учебного взаимодействия в группе (определение общих целей, договор друг с другом и т. д.);

умение критично относиться к своему мнению, с достоинством признавать ошибочность своего мнения (если оно таково) и корректировать его;

умение взглянуть на ситуацию с иной позиции и договариваться с людьми иных позиций.

– *личностные:*

быть мотивированным на учебную деятельность;

воспитание самоконтроля и концентрации, умение правильно распорядиться отведенным временем;

проявление независимости и критичности мышления, воли и настойчивости в достижении цели, творческий подход к решению поставленных задач.

– *предметные:*

В результате изучения программы обучающийся должен

знать:

интерфейс PyScripter и правила написания кода;

синтаксис алгоритмических конструкций на Python;

классические алгоритмы работы с массивами;

терминологию и теорию использования массивов на языке Python;

библиотеки, расширяющие возможности создания Windows-приложений;

приемы работы с виджетами;

типы данных для хранения чисел, строк и способы их преобразования;

уметь:

на языке Python разработать прикладную программу, работающую в операционной системе Windows. Для этого:

- разработать внешний вид программы;
- разработать алгоритм работы программы и реализовать его на Python.

## 1.2. Организационно-педагогические условия реализации программы

*Язык реализации:* русский.

*Форма обучения:* очная.

*Особенности реализации образовательного процесса:*

- *форма реализации образовательной программы:* традиционная с использованием дистанционных технологий;
- *условия набора:* на обучение принимаются все желающие указанной возрастной группы;
- *условия формирования групп:* учащиеся должны быть примерно одного возраста;
- *количество учащихся в группе:* не более 12 человек, что обусловлено возрастными ограничениями по режиму обучения и оснащению рабочих мест;
- *организационная форма обучения:* групповая;
- *формы проведения занятий:* аудиторные (теоретическая и практическая части), онлайн-урок (в период применения дистанционных образовательных технологий);
- *формы организации деятельности учащихся:* фронтальная (объяснение, показ, мастер-класс) и индивидуальная (отработка отдельных навыков).

*Материально-техническое оснащение* программы – рабочее место каждого учащегося оснащено современными техническими средствами с установленным программным обеспечением, объединенным в локальную сеть и имеющим выход в интернет. Программа обеспечена электронными учебно-методическими материалами для учащихся.

*Кадровое обеспечение* – для проведения занятий требуется только педагог дополнительного образования, имеющий профессиональное образование, соответствующее дополнительным общеразвивающим программам.

## 2. Учебный план

№ п/п	Название раздела, темы	Количество часов			Формы контроля/ аттестации
		Всего	Теория	Практика	
1	Базовые конструкции языка Python	39	9,75	29,25	практическая работа
2	Программирование графики с использованием графических модулей Python	12	3	9	практическая работа
3	Создание пользовательского интерфейса с использованием библиотек (Tkinter или Pygame). Разработка игровых приложений	33	8,25	24,75	практическая работа
4	Индивидуальное проектирование "Разработка Windows-приложения на языке Python с использованием библиотеки Tkinter"	24	6	18	защита проекта
	Итого:	108	27	81	

## 3. Календарный учебный график реализации программы «Классическое программирование на языке Python» на 2022/2023 учебный год

Дата начала занятий	Дата окончания занятий	Количество учебных недель	Количество учебных дней	Количество учебных часов	Режим занятий
сентябрь	май	36	36	108	1 раз в неделю по 3 академических часа (1 ак. час – 45 минут)

Продолжительность использования компьютерной техники соответствуют требованиям санитарных правил СП 2.4.3648-20 «Санитарно-эпидемиологические требования к организациям воспитания и обучения, отдыха и оздоровления детей и молодежи».

Календарное планирование годового учебного процесса осуществляется с учетом дат начала занятий и каникулярного периода и реализуется в течение всего учебного года путем последовательного изложения тем в соответствии с Учебным планом.

## 4. РАБОЧАЯ ПРОГРАММА

*С Рабочей программой можно ознакомиться по месту проведения занятий*

### 5. Методические и оценочные материалы

#### 5.1. Методические материалы

Методические материалы программы включают в себя:

- *Методы обучения по источнику передачи и восприятия информации:*
  - словесный: беседа, лекция;
  - наглядный: дидактические и наглядные материал;
  - практический: показ;
- *Методы обучения по характеру деятельности:*
  - объяснительно-иллюстративный (показ, лекция, видеоролик.);
  - репродуктивный (воспроизведение, действие по алгоритму);
  - проектный метод (разработка проектов, создание творческих работ);
  - метод игры (игры дидактические, развивающие, деловые).
- *Активные и интерактивные методы.*
- *Образовательные технологии:*
  - *Педагогические технологии:* индивидуального и группового обучения;
  - *Здоровьесберегающие технологии:*
    - физминутки, зарядки для глаз;
    - смена видов деятельности;
    - дружественная психологическая атмосфера учебного занятия;
    - обучение грамотной заботе о своем здоровье и формированию культуры здоровья учащихся, мотивации их к ведению здорового образа жизни, предупреждению вредных привычек.
- *Формы учебных занятий:*
  - *по особенностям коммуникативного взаимодействия:* лекция, практикум, конкурс, защита;
  - *по дидактической цели:* вводное занятие, занятие по углублению знаний, практическое занятие, занятие по систематизации и обобщению знаний, по контролю знаний, умений и навыков, комбинированные формы занятий.
- *Дидактические материалы:* презентация, видеоролик, рабочая тетрадь, инструкция, задание, пример выполнения задания.

#### 5.2. Оценочные материалы

Для отслеживания результативности освоения образовательной программы проводятся:

- текущий контроль;
- промежуточная аттестация по некоторым разделам.

Текущий контроль успеваемости и качества подготовки учащихся включает фронтальную и индивидуальные формы контроля, проводимые в виде практических и самостоятельных работ, устных опросов, тестирований и педагогических наблюдений.

Обязательные виды текущего контроля в рамках раздела и промежуточной аттестации по некоторым разделам определены календарно-тематическим планом программы, дополнительные – инициируются педагогом с учетом фактического уровня освоения отдельной темы и/или раздела и фиксируются в календарно-тематическом плане рабочей программы конкретной группы.

Критерии оценивания по всем видам текущего контроля, кроме педагогического наблюдения, и промежуточной аттестации основываются на следующих правилах:

Процент выполнения заданий контроля	Уровень освоения
>80%	отличный
от 60% до 80% включительно	хороший
от 30% до 60% включительно	удовлетворительный
<30%	неудовлетворительный

Критерии оценивания посредством педагогического наблюдения основываются на следующих правилах:

Факт выполнения в соответствии с индивидуальными особенностями учащегося	Результат освоения
выполнил	зачтено
не выполнил	не зачтено

Оценка результативности обучения по разделам, где не определена промежуточная аттестация, производится по результатам текущего контроля, а при отсутствии таковых – по результатам педагогического наблюдения.

Общая оценка результативности обучения по программе базируется на освоении отдельных разделов не ниже удовлетворительного уровня.

В пунктах 5.2.1-5.2.4 представлены примеры заданий для проведения текущего контроля и промежуточных аттестаций по некоторым разделам программы.

5.2.1. Задания для промежуточной аттестации по разделу 1 «Базовые конструкции языка Python»

Пример самостоятельной работы по теме «Решение практических задач с использованием циклов»

*Вариант 1*

*Задание 1.* В области 12 районов. Известны количество жителей (в тысячах человек, (10000-100000) и площадь построек (в км<sup>2</sup>, площадь от 1000 до 2000) каждого района. Определить среднюю плотность населения по области в целом и каждого района. Решить задачу, используя циклическую конструкцию `for` и случайные числа.

*Задание 2.* Мой богатый дядюшка подарил мне один доллар в мой первый день рождения. В каждый день рождения он удваивал свой подарок и прибавлял к нему столько долларов, сколько лет мне исполнилось. Написать программу, указывающую, к какому дню рождения подарок превысит 100\$. Решить задачу используя циклическую конструкцию `while`.

*Вариант 2*

*Задание 1.* Начав тренировки, лыжник в первый день пробежал 10 км. Каждый следующий день он увеличивал пробег на 10% от пробега предыдущего дня.



Определить:

- а) пробег лыжника за второй, третий, ..., десятый день тренировок;
- б) какой суммарный путь он пробежал за первые 7 дней тренировок.

Решить задачу используя циклическую конструкцию for.

*Задание 2.* Найти сумму и произведение цифр, введенного целого числа. Например, если введено число 325, то сумма его цифр равна 10 ( $3+2+5$ ), а произведение 30 ( $3*2*5$ ). Решить задачу используя циклическую конструкцию while.

Пример контрольной работы по теме  
«Массивы. Одномерные массивы (списки). Типовые алгоритмы обработки списков»

*Вариант 1*

*Задание 1.* Дан целочисленный массив из 31 элемента, в котором записаны значения температуры воздуха в марте. Элементы массива могут принимать значения от  $(-20)$  до 20. Программа должна вывести среднюю температуру по всем дням, когда была оттепель (температура поднималась выше нуля). Гарантируется, что, хотя бы один день в марте была оттепель.

*Задание 2.* Дано целое число ( $N > 0$ ). Если оно является степенью числа 3, то вывести True, если не является — вывести False. Использовать процедуру.

*Задание 3.* Даны действительное число  $a$ , натуральное число  $n$ . Составьте программу, которая вычисляет произведение  $a*(a+1)*...*(a+n-1)$ . Числа  $a$  и  $n$  введите с клавиатуры.

*Вариант 2*

*Задание 1.* Дан целочисленный массив из 30 элементов. Элементы массива могут принимать целые значения от 0 до 100 – баллы учащихся выпускного класса, полученные на экзамене по информатике. Напишите программу, которая позволяет найти и вывести количество учащихся, чьи баллы на экзамене выше среднего балла по классу.

*Задание 2.* Дано  $n$ -значное число. Посчитать количество цифр 5 в этом числе. Если их нет, вывести сообщение «Нет в числе». Использовать процедуру.

*Задание 3.* Дан массив из 30 элементов. Каждый элемент может принимать целочисленные значения от 0 до 1000. Составить программу, которая позволит найти сумму двузначных четных элементов массива.

Результаты выполнения работы отражаются в Журнале учёта успеваемости.

Пример самостоятельной работы по темам  
«Строковые переменные. Операции со строками», «Работа с файлами в Python»

*Задание 1.* Задание 1. Вывести на экран все слова, состоящие из  $L$  букв, которые можно построить из букв некоторого алфавита ( $A$ ), где  $A$  – строка, состоящая из букв алфавита.

*Задание 2.* Задание 2. Отсортировать по алфавиту список фамилий.

*Задание 3.* Задание 3. Вводится строка, состоящая из букв и пробелов. Составить из входящих в нее букв несколько сочетаний (слов) любой длины. Каждую букву можно использовать любое количество раз.

*Задание 4.* Задание 4. Определить самое длинное слово в строковом массиве и его номер.

Результаты выполнения работы отражаются в Журнале учёта успеваемости.

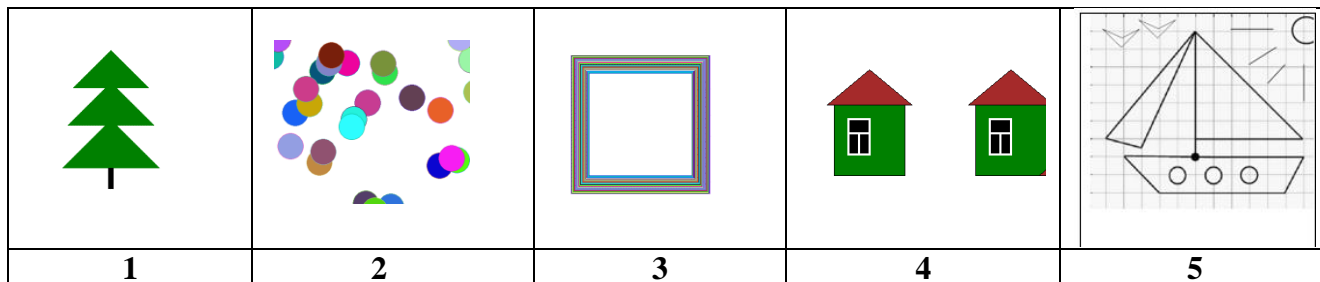
5.2.2 Задание для промежуточной аттестации  
по разделу 2 «Программирование графики с использованием графических модулей Python»

Задача 1. Нарисовать елку (использовать процедуру). (рис.1)

Задача 2. Построить окружности случайного цвета и случайного радиуса (рис.2)

Задача 3. Построить квадраты случайного цвета (рис.3)

Задача 4. Построить рисунки 4-5:



5.2.3. Задания для промежуточной аттестации по разделу 3 «Создание пользовательского интерфейса с использованием библиотек (Tkinter или Pygame). Разработка игровых приложений»

Примеры самостоятельных работ по созданию пользовательского интерфейса с использованием библиотеки Tkinter

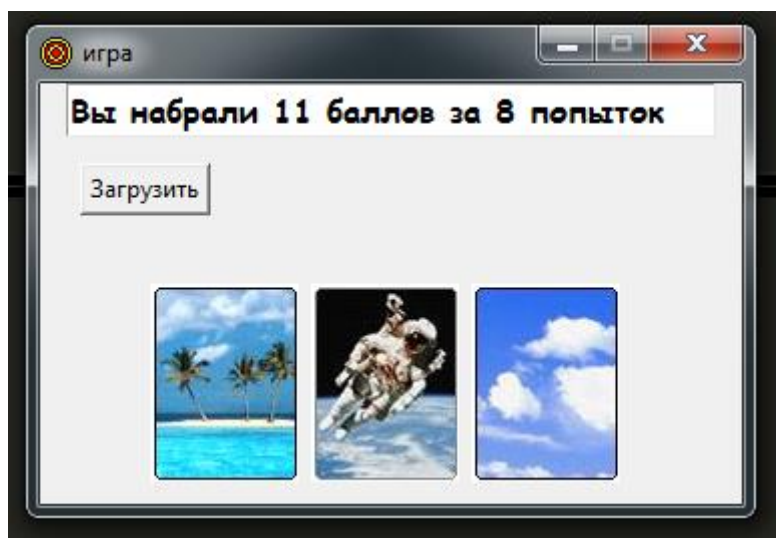
Самостоятельная работа 1.

Проверка умения настраивать и размещать виджеты Button и Entry и способов добавления изображений для создания интерфейса приложения.

Задание 1. Разработать программу «Калькулятор», использовать процедуры.



Задание 2. Написать скрипт для создания игрового приложения:



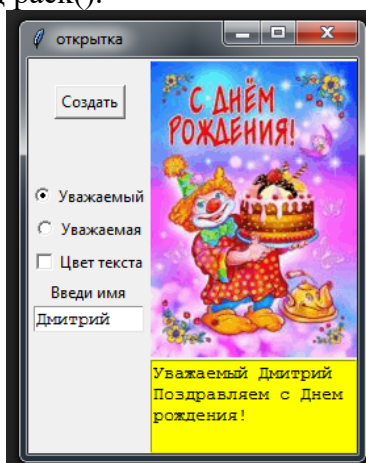
- Создать форму, для формы установить размеры, расположение на экране, иконку, запретить изменять размеры окна по ширине и высоте.
- Разместить: кнопку, 3 метки для размещения картинок, текстовое поле.
- По кнопке Загрузить в метки загружаются картинки, имена файлов с картинками записываем в массив. Картинки в формате gif. Картинки выбираются из массива случайным образом, их всего 6.
- Для загрузки изображений использовать процедуру
- Нужно посчитать количество баллов и количество попыток и результат выводить в текстовое поле
- Если выпадают все картинки разные, то количество баллов увеличивается на 1
- Если выпадает 2 одинаковые картинки, то количество баллов увеличивается на 2
- Если выпадает 3 одинаковые картинки, то количество баллов увеличивается на 3
- Количество попыток – 10, после вывода результата десятой попытки форма закрывается.

Результаты выполнения работы отражаются в Журнале учёта успеваемости.

## Самостоятельная работа 2.

Проверка умения использования виджетов различного назначения для создания интерфейса приложения

Задание 1. Разработать программу «Поздравительная открытка», для размещения объектов использовать фреймы, метод pack().



Задание 2. Разработать программу, скриншот в задании, разместить 2 флажка и метку с картинкой. При изменении состояния первого флажка меняется цвет формы, изменение состояния второго флажка влечет изменение картинки.

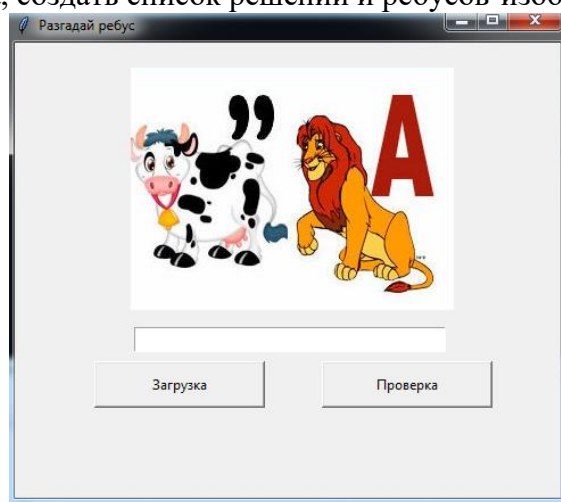


Результаты выполнения работы отражаются в Журнале учёта успеваемости.

### Самостоятельная работа 3.

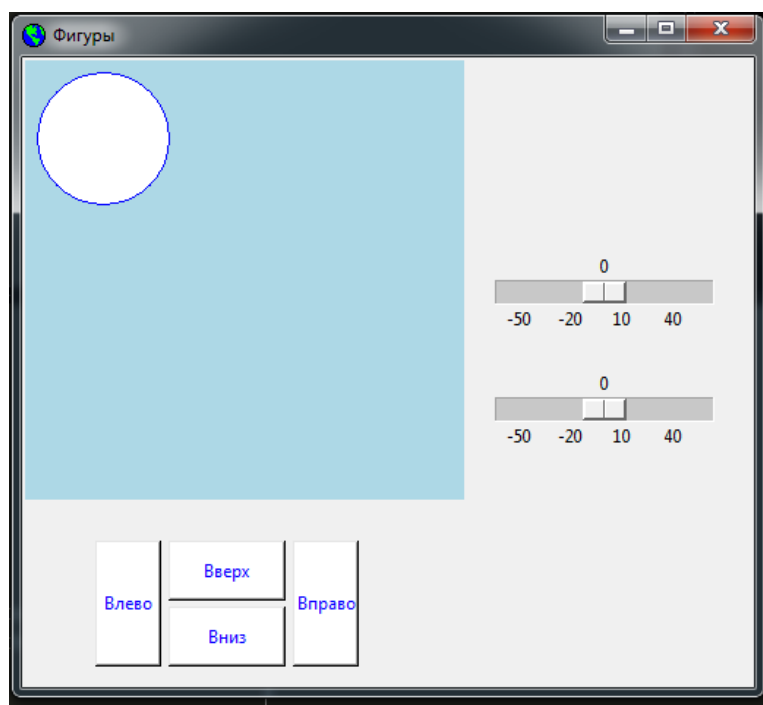
Проверка умения использования виджетов различного назначения для создания интерфейса приложения

Задание 1. Разработать программу, скриншот в задании, разместить 2 кнопки, метку для изображений, виджет text, создать список решений и ребусов-изображений, посчитать баллы.



Задание 2. Разработать приложение по образцу на скриншоте.

- Создать форму, загрузить иконку, установить размеры (500x430), запретить изменять размеры по ширине и высоте.
- Разместить на форме полотно - объект класса Canvas, 2 шкалы для изменения ширины и высоты фигуры, размещенной на полотне (окружность), цвет полотна, обводки и заливки задать на свой вкус.
- Разместить 4 кнопки, с помощью которых можно менять положение круга на полотне (влево, вправо, вверх, вниз).



Результаты выполнения работы отражаются в Журнале учёта успеваемости.

5.2.4. Задание для промежуточной аттестации  
по разделу 4 «Проектная работа "Разработка Windows-приложения на языке Python с использованием библиотеки Tkinter"»

Пример задания на проектную работу

*Постановка задачи:*

1. Согласно индивидуальному заданию создать приложение на языке Питон.
2. Составить пояснительную записку к проекту.

*Общие требования к выполнению работы:*

1. Проект должен быть сохранен в папку Z:\PP\_Itoг\смысловое имя\_Фамилия\_Группа.
2. В папке Img должны находиться файлы ресурсов (если нужны). Лишние файлы должны быть удалены.
3. Код должен компилироваться, и программа должна запускаться на выполнение.
4. Демоверсия отражает только основной минимальный функционал, который обязательно должен быть выполнен. Для получения отличной оценки проект должен содержать творческие дополнения на вкус разработчика.
5. Все имена переменных и элементов управления должны быть осмысленными.
6. Код вашей программы должен быть структурированным.
7. Код должен быть экономичным, использование ресурсов компьютера оправданным.
8. При вводе каких-либо данных в вашу программу вы должны предусмотреть проверки корректности ввода всех данных.
9. Все элементы должны быть аккуратно расположены на форме. Назначение тех или иных элементов должно быть ясно пользователю.
10. Приложение должно иметь адекватное дизайнерское решение. Рекомендуется применение элементов оформления в виде изображений, фоновых рисунков или заливок. Все формы должны иметь соответствующий заголовок и иконку.
11. При запуске вашего проекта все ненужные элементы управления должны быть заблокированы. По мере работы с программой пользователь должен иметь возможность работать только с теми ЭУ, которые необходимы в данный момент, и никакие другие.
12. Пояснительная записка должна содержать:
  - описание работы проекта (для пользователя);
  - скриншоты форм с указанием измененных свойств элементов управления;
  - код программы с пояснениями.
13. Список заданий:
  - 1) Орфографическая викторина
  - 2) Викторина любого тапа
  - 3) Графический редактор
  - 4) Словарь английского языка
  - 5) Текстовый редактор
  - 6) Сборник игровых заданий для детей
  - 7) Калькулятор
  - 8) Гороскоп
  - 9) Флаги стран
  - 10) Память (парные картинки)
  - 11) Игра «Поле чудес»
  - 12) Игра «Виселица»
  - 13) Произвольное задание

**5.3. Оценка удовлетворенности образовательным процессом**

*5.3.1. Анкета для учащихся  
дополнительной общеразвивающей программы  
«Классическое программирование на языке Python»*

№	Вопросы	да	нет	не всегда
1	Я иду на занятия с радостью			
2	К педагогу можно обратиться за советом и помощью в трудной ситуации			
3	На занятиях я могу всегда свободно высказать своё мнение			
4	Учебные материалы, используемые на занятиях, интересны и полезны			
5	Учебные кабинеты Академии имеют хорошее техническое оснащение			
6	В Академии создаются условия, чтобы учиться было комфортно			
7	В Академии я испытываю уважительное отношение со стороны педагогов			
8	На занятиях педагог обращается ко мне по имени			
9	Академия для меня - безопасное место, где я комфортно чувствую себя			
10	Моя успеваемость в школе повысилась благодаря занятиям в Академии			
11	Я уверен в своих знаниях по изучаемому в Академии предмету			
12	Я мог бы рекомендовать занятия в Академии своим знакомым			
13	Преподаватель замечает мои успехи и хвалит			
14	Я горжусь тем, что обучаюсь в Академии			

*5.3.2. Анкета для родителей учащихся  
дополнительной общеразвивающей программы  
«Классическое программирование на языке Python»*

№	Критерий качества образования	Полностью удовлетворён	Затрудняюсь ответить	Не удовлетворён
1	Удовлетворены ли Вы качеством образовательных услуг?			
2	Удовлетворены ли Вы компетентностью педагогов Академии?			
3	Оцените доброжелательность и вежливость педагогов			
4	Считаете ли Вы, что образовательный процесс в Академии ориентирован на развитие личности ребенка?			
5	Оцените качество взаимодействия с администрацией Академии			
6	Удовлетворены ли Вы материально-техническим обеспечением Академии?			
7	Готовы ли Вы рекомендовать Академию друзьям и знакомым?			

## 6. Литература и учебные пособия

1. Описание модулей Tkinter: [Электронный ресурс]: <https://www.russianlutheran.org/python/life/life.htm> .
2. [Электронный ресурс]: <https://pythonworld.ru/samouchitel-python>
3. [Электронный ресурс]: <https://www.russianlutheran.org/python/life/life.htm>
4. [Электронный ресурс]: <https://web-start.top/ru/progru/pythonru/pygameru>
5. [Электронный ресурс]: <https://younglinux.info>
6. Уоррен Сэнд, Картер Сэнд, «Hello World! Занимательное программирование», Издательский дом «Питер», 2020.-400с.
7. Пол Бэрри. Изучаем программирование на Python (перевод с англ. М.А.Райтмана) - Москва: Изд-во «Э», 2021. – 624с.
8. Майк Макграт. Python. Программирование для начинающих. (перевод с англ. М.А.Райтмана) - Москва, «Эксмо», 2020. – 192с.
9. Эл Свейгарт. Большая книга проектов Python. СПб: «Питер», 2022. – 432с.