

МКУ Отдел образования Администрации Парабельского района

Муниципальное бюджетное общеобразовательное учреждение
«Парабельская средняя школа имени Николая Андреевича Образцова»

Центр цифрового и гуманитарного профилей «Точка роста»

Принята на заседании
педагогического совета
от «02» сентября 2024г.
Протокол № 10

Утверждаю:
и.о. директора
Петрова М.И./
Приказ № 239
от «02» сентября 2024г.



Дополнительная общеобразовательная общеразвивающая программа
технической направленности
«Программирование на Python»
Возраст обучающихся: 15 – 18 лет
Срок реализации: 1 год

Составитель:
Коваленко Тамара Анатольевна,
педагог дополнительного образования

с.Парабель

2024

Пояснительная записка

Дополнительная общеобразовательная общеразвивающая программа **«Программирование на Python»** имеет **техническую** направленность.

В современную жизнь человека все шире внедряются компьютеры и информационные технологии. Поэтому все большее значение приобретает компьютерная грамотность.

Изучение компьютерной грамотности состоит из двух разделов: пользовательского курса и программирования. Раздел «Программирование» в школьном курсе представлен языком программирования Pascal, а многим учащимся хочется познакомиться с другими языками программирования, самим попробовать разработать программы, которые можно использовать на уроках и во внеурочной деятельности. Данная программа позволяет реализовать эти желания, так как уделяется большое внимание практической работе учащихся на компьютере, самостоятельной разработке ими программ для решения практических задач.

Актуальность программы

С развитием современных информационных технологий сегодня любой учащийся под руководством опытного педагога может с лёгкостью научиться программировать.

Компьютеры и компьютерные системы – неотъемлемая часть жизни нашего общества. Научившись программировать, мы можем быть не только пользователями информационных технологий, но и активными их создателями.

Языки программирования можно сравнить с иностранными языками, овладеть ими может каждый. Учиться программировать очень интересно. Результат программирования очень часто виден сразу. Кроме того, создание компьютерных игр и обучающих программ способствует развитию логики и креативного мышления. Ещё одной значимой стороной обучения программированию является спрос на рынке труда на специалистов данного направления деятельности.

Новизна программы состоит в том, что в программе использована технология проектного обучения для формирования предметных навыков в области информационных технологий.

Отличительные особенности

Основное количество часов отводится практическому написанию программ. Каждый обучающийся реализует индивидуальный проект в результате освоения программы. Продукт, полученный в результате освоения программы, имеет прикладной характер и может быть использован по необходимости.

Цель: формирование интереса к техническим видам творчества, развитие логического, технического мышления, создание условий для творческой самореализации личности ребёнка посредством получения навыков разработки эффективных алгоритмов, для реализации их в виде программы, написанной на языке программирования Python.

Задачи:

Обучающие:

- обучить языку программирования Python и созданию программ на его основе;
- научить создавать прикладное программное обеспечение;
- расширять кругозор обучающихся в области программирования;
- научить дизайнерскому оформлению созданного ПО.

Развивающие:

- развивать память и внимание, познавательную и творческую активность;
- развивать творческие способности, эстетическое и эргономическое восприятие объектов труда;
- развивать логическое мышление.

Воспитательные:

- прививать интерес к активному творческому самовыражению, культуре труда;
- воспитывать упорство в достижении желаемого результата;
- воспитывать эстетический вкус;
- воспитывать чувство взаимопомощи, доверия, коллективизма.

Адресат программы: обучающиеся от 15 до 18 лет. Наполняемость группы до 15 человек.

Объем и срок реализации: программа рассчитана на 1 год. Общее количество часов в год составляет 68 часов.

Форма обучения: очная.

Режим занятий: занятия проводятся 1 раз в неделю по 2 часа.

Дополнительная общеразвивающая программа «Программирование на Python» разработана на основе следующих **нормативных правовых документов:**

- Федеральный Закон от 29.12.2012 № 273-ФЗ «Об образовании в РФ».
- Постановление Главного государственного санитарного врача РФ от 28.08.2020г. № 28 «Об утверждении санитарных правил СП 2.4. 3648-20 «Санитарно-эпидемиологические требования к организациям воспитания и обучения, отдыха и оздоровления детей и молодежи».
- Письмо Министерства образования и науки РФ от 18.11.2015 г. № 09-3242 «О направлении информации: методические рекомендации по проектированию дополнительных общеразвивающих программ».
- Приказ Министерства просвещения РФ от 27.07.2022г. №629 «Об утверждении Порядка организации и осуществления образовательной деятельности по дополнительным общеобразовательным программам».
- Концепция развития дополнительного образования детей до 2030 года (утверждена распоряжением Правительства Российской Федерации от 31.03.2022 г. № 678-р)

Планируемые результаты

Предметные результаты:

- умение определять результат выполнения алгоритма при заданных исходных данных, узнавать изученные алгоритмы обработки чисел и числовых последовательностей, создавать на их основе несложные программы анализа данных, читать и понимать несложные программы, написанные на выбранном для изучения универсальном алгоритмическом языке высокого уровня;
- понимание основных предметных понятий («информация», «алгоритм», «исполнитель», «модель») и их свойств;
- развитие логических способностей и алгоритмического мышления, умения составить и записать алгоритм для конкретного исполнителя, знакомство с основными алгоритмическими структурами – линейной, условной и циклической;
- умение выполнять пошагово (с использованием компьютера или вручную) несложные алгоритмы управления исполнителями и анализа числовых и текстовых данных;
- навыки и опыт разработки программ в выбранной среде программирования, включая тестирование и отладку программ; умение использовать основные управляющие конструкции объектно-ориентированного программирования и библиотеки прикладных программ, выполнять созданные программы;
- умение разрабатывать и использовать компьютерно-математические модели, оценивать числовые параметры моделируемых объектов и процессов, 18 интерпретировать результаты, получаемые в ходе моделирования реальных процессов, анализировать готовые модели на предмет соответствия реальному

объекту или процессу;

- умение соблюдать нормы информационной этики и права.

Личностные результаты:

- формирование ответственного отношения к учению, способности довести до конца начатое дело аналогично завершённым творческим учебным проектам;
- формирование способности к саморазвитию и самообразованию средствами информационных технологий на основе приобретённой благодаря иллюстрированной среде программирования мотивации к обучению и познанию;
- развитие опыта участия в социально значимых проектах, повышение уровня самооценки благодаря реализованным проектам;
- формирование коммуникативной компетентности в общении и сотрудничестве со сверстниками в процессе образовательной, учебно-исследовательской и проектной деятельности, участия в конкурсах и конференциях различного уровня;
- формирование целостного мировоззрения, соответствующего современному уровню развития информационных технологий;
- формирование осознанного позитивного отношения к другому человеку, его мнению, результату его деятельности;
- формирование ценности здорового и безопасного образа жизни; усвоение правил индивидуального и коллективного безопасного поведения при работе с компьютерной техникой.

Метапредметные результаты:

- умение самостоятельно ставить и формулировать для себя новые задачи, развивать мотивы своей познавательной деятельности;
- умение самостоятельно планировать пути решения поставленной проблемы для получения эффективного результата; понимание, что в программировании длинная программа не всегда лучшая;
- умение критически оценивать правильность решения учебно- исследовательской задачи;
- умение корректировать свои действия, вносить изменения в программу и отлаживать её в соответствии с изменяющимися условиями;
- владение основами самоконтроля, способность к принятию решений;
- умение создавать, применять и преобразовывать знаки и символы, модели и схемы для решения учебно-исследовательских и проектных работ;
- формирование и развитие компетентности в области использования информационно-коммуникационных технологий (ИКТ-компетенция);
- умение организовывать учебное сотрудничество и совместную деятельность с учителем и сверстниками в процессе проектной и учебно-исследовательской деятельности.

Формы аттестации

Результаты освоения программы отслеживаются по итогам опросов, выполнения практических заданий.

Формы подведения итогов реализации программы

Результаты обучения по программе выявляются по итогам разработки и защиты проекта.

Учебно-тематический план

№ /п	Наименование раздела, темы	Количество часов			Формы контроля
		Всего	Теория	Практика	

1	Введение в Python	7	3	4	
1.1	Язык Python. Среда программирования. Особенности ввода-вывода	2	1	1	Практическое задание
1.2	Типы данных, операции. Оператор присваивания	2	1	1	Практическое задание
1.3	Числа. Стандартные операции	3	1	2	Практическое задание
2	Алгоритмические конструкции	11	3	8	
2.1	Условный оператор	1	1	-	Опрос
2.2	Цикл while.	5	1	4	Практическое задание
2.3	Цикл for	5	1	4	Практическое задание
3	Строки	10	4	6	
3.1	Литералы строк	2	1	1	Практическое задание
3.2	Срезы строк	2	1	1	Практическое задание
3.3	Методы строк	6	2	4	Практическое задание
4	Функции	10	4	6	
4.1	Парадигма и преимущество структурного программирования	1	1	-	Опрос
4.2	Граф и стек вызовов функций. Области видимости переменных	3	1	2	Практическое задание
4.3	Прямая рекурсия	3	1	2	Практическое задание
4.4	Косвенная рекурсия	3	1	2	Решение задач повышенной сложности
	Списки и кортежи	10	4	6	
5.1.	Списки и кортежи в Python. Сходства и различия	1	1	-	Опрос
5.2	Операции со списками	5	1	4	Практическое задание
5.3	Срезы списков	2	1	1	Практическое задание
5.4	Матрицы. Операции над матрицами	2	1	1	Практическое задание
6	Слова и множества	4	2	2	
6.1	Словари	2	1	1	Практическое задание
6.2	Множества	2	1	1	Практическое задание

7	Объектно-ориентированное программирование	16	2	14	
7.1	Классы в Python	1	1	-	Опрос
7.2	Разработка собственного класса	3	1	2	Практическое задание
7.3	Разработка и программирование собственного проекта	12	-	12	
	Всего:	68	22	46	

Календарный учебный график

№ п/п	Дата	Форма занятия	Наименование раздела, темы	Кол-во часов	Формы контроля
Введение в Python, 7 часов					
1		Видеозанятие	Язык Python. Среда программирования. Особенности ввода-вывода	1	Опрос
2		Практическое занятие		1	Практическое задание
3		Видеозанятие	Типы данных, операции. Оператор присваивания	1	Опрос
4		Практическое занятие		1	Практическое задание
5		Видеозанятие	Числа. Стандартные операции	1	Опрос
6		Практическое занятие		1	Практическое задание
7		Практическое занятие		1	Практическое задание
Алгоритмические конструкции, 11 часов					
8		Видеозанятие	Условный оператор	1	Опрос
9		Практическое занятие	Цикл while.	1	Практическое задание
10		Практическое занятие		1	Практическое задание
11		Практическое занятие		1	Практическое задание
12		Практическое занятие		1	Практическое задание
13		Практическое занятие		1	Практическое задание
14		Видеозанятие	Цикл for	1	Опрос
15		Практическое занятие		1	Практическое задание
16		Практическое занятие		1	Практическое задание
17		Практическое занятие		1	Практическое задание

17		Практическое занятие		1	Практическое задание
Строки, 10 часов					
19		Видеозанятие	Литералы строк	1	Опрос
20		Практическое занятие		1	Практическое задание
21		Видеозанятие	Срезы строк	1	Опрос
22		Практическое занятие		1	Практическое задание
23		Видеозанятие	Методы строк	1	Опрос
24		Практическое занятие		1	Практическое задание
25		Практическое занятие		1	Практическое задание
26		Практическое занятие		1	Практическое задание
26		Практическое занятие		1	Практическое задание
28		Практическое занятие		1	Практическое задание
Функции, 10 часов					
29		Видеозанятие	Парадигма и преимущество структурного программирования	1	Опрос
30		Видеозанятие	Граф и стек вызовов функций. Области видимости переменных	1	Опрос
31		Практическое занятие		1	Практическое задание
32		Практическое занятие		1	Практическое задание
33		Видеозанятие	Прямая рекурсия	1	Опрос
34		Практическое занятие		1	Практическое задание
35		Практическое занятие		1	Практическое задание
36		Видеозанятие	Косвенная рекурсия	1	Опрос
37		Практическое занятие		1	Решение задач повышенной сложности
38		Практическое занятие		1	Решение задач повышенной сложности
Списки и кортежи, 10 часов					
39		Видеозанятие	Списки и кортежи в Python. Сходства и различия	1	Опрос
40		Видеозанятие	Операции со списками	1	Опрос
41		Практическое занятие		1	Практическое задание
42		Практическое занятие		1	Практическое задание

43		Практическое занятие		1	Практическое задание
44		Практическое занятие		1	Практическое задание
45		Видеозанятие		Срезы списков	1
46		Практическое занятие	1		Практическое задание
47		Видеозанятие	Матрицы. Операции над матрицами	1	Опрос
48				1	Практическое задание
Слова и множества, 4 часа					
49		Видеозанятие	Словари	1	Опрос
50		Практическое занятие		1	Практическое задание
51		Видеозанятие	Множества	1	Опрос
52		Практическое занятие		1	Практическое задание
Объектно-ориентированное программирование, 16 часов					
53		Видеозанятие	Классы в Python	1	Опрос
54		Видеозанятие	Разработка собственного класса	1	Опрос
55		Практическое занятие		1	Практическое задание
56		Практическое занятие		1	Практическое задание
57		Видеозанятие	Разработка и программирование собственного проекта	1	Опрос
58		Практическое занятие		1	Практическое задание
59		Практическое занятие		1	Практическое задание
60		Практическое занятие		1	Практическое задание
61		Практическое занятие		1	Практическое задание
62		Практическое занятие		1	Практическое задание
63		Практическое занятие		1	Практическое задание
64		Практическое занятие		1	Практическое задание
65		Практическое занятие		1	Практическое задание
66		Практическое занятие		1	Практическое задание
67		Практическое занятие		1	Защита проекта
68		Практическое занятие		1	Защита проекта

Содержание программы

1. Введение в Python

Основные понятия: трансляция, интерпретация, компиляция, синтаксис, семантика, прагматика, переменная, динамическая типизация, служебные слова, идентификаторы, простые типы данных, приоритеты операций, литералы чисел, операция присваивания, PEP 8.

1.1 Язык Python. Среда программирования. Особенности ввода- вывода

Теория. Язык программирования Python. Достоинства и недостатки. Области применения. Интерактивный режим работы программы.

Практика. Установка языка программирования Python 3.9 и среды программирования IDLE.

1.1 Типы данных, операции. Оператор присваивания

Теория. Ввод и вывод числовой информации.

Практика. Тренировочное задание на ввод и вывод числовой информации.

1.2 Числа. Стандартные операции

Теория. Стандартные операции с целыми и действительными числами. Стиль программирования Python.

Практика. Решение простых задач в интерактивном режиме.

2. Алгоритмические конструкции

Основные понятия: логический тип данных, логические операции (and, or, not), условный оператор, условное и альтернативное исполнение алгоритма, операторы сравнения, вложенность операторов, оператор цикла, переменная-флаг, генерация псевдослучайной последовательности, инструкции break, continue и pass.

2.1 Условный оператор

Теория. Условная и циклическая инструкции. Каскадность и вложенность алгоритмических инструкций.

2.2 Цикл while

Теория. Теория чисел. Фильтрация потока чисел. НОД и НОК. Проверка числа на простоту. Алгоритм Евклида. Нахождение максимума и минимума.

Практика. Решение задач на анализ чисел потока и целочисленной арифметики.

2.2 Цикл for

Теория. Теория чисел. Фильтрация потока чисел. НОД и НОК. Проверка числа на простоту. Алгоритм Евклида. Нахождение максимума и минимума.

Практика. Решение задач на анализ чисел потока и целочисленной арифметики повышенной трудности.

3. Строки

Основные понятия: символ, строка, литерал, таблицы кодов ASCII, UTF-8, отладка кода, неизменяемый объект, формат вывода строки, экранированные escape-последовательности, положительная и отрицательная нумерация символов в строке, срез, конкатенация, длина строки.

3.1 Литералы строк

Теория. Понятие «литералы строк».

Практика. Ввод-вывод строки. Решение задач на ввод строки, поиск подстроки.

3.2 Срезы строк

Теория. Форматирование строки.

Практика. Преобразование строки. Решение задач.

3.3 Методы строк

Теория. Методы работы со строкой.

Практика. Применение методов строки. Решение задач.

4. Функции

Основные понятия: подпрограмма, функция, процедура, рекурсия, глубина рекурсии,

объявление, определение и вызов функции, возврат значений, глобальные и локальные переменные, передача параметров, работа с памятью, граф вызовов, стек вызовов, полиморфизм функций, утиная типизация.

4.1 Парадигма и преимущества структурного программирования

Теория. Обзор парадигм программирования. Особенности применения языков программирования.

4.2 Граф и стек вызовов функций. Области видимости переменных

Теория. Применение стек и граф вызовов функций.

Практика. Выполнение тренировочных упражнений.

4.3 Прямая рекурсия

Теория. Понятие прямой рекурсии и ее применение.

Практика. Выполнение тренировочных упражнений.

4.4 Косвенная рекурсия

Теория. Понятие косвенной рекурсии и ее применение.

Практика. Выполнение тренировочных упражнений. Практические занятия к темам 4.2., 4.3., 4.4.: Нахождение суммы чисел. Числа Фибоначчи. Вычисление степени. Ханойские башни. Использование библиотеки математических функций. Решение задач повышенной трудности.

5. Списки и кортежи

Основные понятия: список, кортеж, элемент списка и кортежа, индекс, срез списка, матрица, многомерный список, сортировка, сложность алгоритма, устойчивость сортировки, квадратичная, быстрая, синхронная, поразрядная сортировки списка, случайное перемешивание.

5.1 Списки и кортежи в Python. Сходства и различия

Теория. Представление списка и кортежа в памяти компьютера, сходства и различия.

5.2 Операции со списками

Теория. Способы заполнения списка (с клавиатуры, из файла, случайным образом, по формуле).

Практика. Решение задач на ввод-вывод элементов одномерного и многомерного списка.

5.3 Срезы списков

Теория. Методы работы со списком и кортежем. Методы сортировки списка.

Практика. Решение задач на ввод-вывод элементов одномерного и многомерного списка и кортежа, преобразование, поиск, замену, подсчет.

5.4 Матрицы. Операции над матрицами

Теория. Вычисление сложности алгоритма. Многомерные списки.

Практика. Решение задач повышенной трудности.

6. Словари и множества

Основные понятия: словарь, множество, ключ, кодирование.

6.1 Словари

Теория. Понятие словаря. Способы создания. Словарь, преимущества и недостатки, методы работы со словарем. Словари со смешанными значениями. Кодирование и декодирование текста.

Практика. Решение задач на заполнение, преобразование, поиск, замену, подсчет, вывод элементов словаря.

6.2 Множества

Теория. Понятие множества. Создание множеств. Множество, преимущества и недостатки, методы работы с множеством.

Практика. Решение задач повышенной трудности.

7. Объектно-ориентированное программирование (ООП)

Основные понятия: ООП, класс, метод INIT, экземпляр, наследование, полиморфизм, исключения, виджет, интерфейс, событие, техническое задание, проект, проектная деятельность, виды проектов.

7.1 Классы в Python

Теория. Понятия «класс», «метод INIT», «экземпляр», «наследование», «полиморфизм», «исключения», «виджет», «интерфейс», «интерфейс», «событие».

7.2 Разработка собственного класса

Теория. Принципы разработки собственного класса. Обработка и генерация исключений. Виджет, методы виджета. Графическая библиотека tkinter, класс Tk. Системные методы.

Практика. Создание собственного класса.

7.3. Разработка и программирование собственного проекта

Практика. Выбор вида и темы проекта. Составление технического задания.

Программирование. Разработка технической документации и презентации проекта.

Методическое обеспечение и условия реализации Программы

Материально-техническое обеспечение:

- компьютер для демонстрации презентаций;
- проектор;
- рабочие компьютеры учащихся для работы с доступом в Интернет;
- принтер для распечатки заданий
- распечатки заданий для практикумов;
- презентационные материалы для объяснения;
- карточки с индивидуальными заданиями.

Список литературы

1. Доусен М. Програмуємо на Python / М. Доусен - СПб.: Питер, 2016. - 416с.
2. Лутц М. Изучаем Python, 4 издание / М. Лутц - СПб.: Символ-Плюс, 2011. - 1280 с.
3. Любанович Б. Простой Python. Современный стиль программирования / Б. Любанович. - СПб.: Питер, 2016. - 480с.
4. Прохоренок Н.А., Дронов В.А. Python 3 и PyQt 5. Разработка приложений / Н.А. Прохоренок, В.А. Дронов - СПб.: «БХВ-Петербург», 2016. - 832с.
5. Саммерфильд М. Python на практике / М. Саммерфильд, пер. А.А.Слинкин – М.: ДМК-Пресс, 2014. - 338с.

Цифровые образовательные ресурсы

1. <https://pythontutor.ru/>
2. <https://www.python.org/>
3. <https://pythonworld.ru/samouchitel-python>
4. <https://pythoner.name/>

Формы аттестации и оценочные материалы

Устный опрос, анкетирование, тестирование, наблюдение на протяжении обучения, самостоятельная работа.

Формы контроля успешности обучающихся и подведения итогов реализации программы:

- Результативность работы планируется отслеживать в течение учебного года на занятиях путем педагогического наблюдения (развитие каждого ребенка и группы в целом).
- Текущий контроль предполагается проводить на каждом занятии – подведение итогов с перспективой на будущее, диалоги, игры на развитие логики, внимания, памяти.
- Промежуточный контроль проводится после изучения каждой темы – обобщающее повторение (проведение тестов на знание теоретического материала и практические задания).
- Итоговый контроль предполагает анализ усвоения образовательной программы обучающимися.

Приложение 1

Требования техники безопасности труда

Персональный компьютер – это электроприбор. От прочих электроприборов он отличается тем, что для него предусмотрена возможность длительной эксплуатации без отключения от электрической сети. Кроме обычного режима работы компьютер может находиться в режиме работы с пониженным электропотреблением или в дежурном режиме ожидания запроса. В связи с возможностью продолжительной работы компьютера без отключения от электросети следует уделить особое внимание качеству организации электропитания.

1. Недопустимо использование некачественных и изношенных компонентов в системе электроснабжения, а также их суррогатных заменителей: розеток, удлинителей, переходников, тройников. Недопустимо самостоятельно модифицировать розетки для подключения вилок, соответствующих иным стандартам. Электрические контакты розеток не должны испытывать механических нагрузок, связанных с подключением массивных компонентов (адаптеров, тройников и т.п.)

2. Все питающие кабели и провода должны располагаться с задней стороны компьютера и периферийных устройств. Их размещение в рабочей зоне пользователя недопустимо.

3. Запрещается производить какие-либо операции, связанные с подключением, отключением или перемещением компонентов компьютерной системы без предварительного отключения электропитания.

4. Компьютер не следует устанавливать вблизи электронагревательных приборов и систем отопления.

5. Недопустимо размещать на системном блоке, мониторе и периферийных устройствах посторонние предметы: книги, листы бумаги, салфетки, чехлы от пыли. Это приводит к постоянному или временному перекрытию вентиляционных отверстий. 6. Запрещается внедрять посторонние предметы в эксплуатационные или вентиляционные отверстия компонентов компьютерной системы.

7. Монитор имеет элементы, способные сохранять высокое напряжение в течение длительного времени после отключения от электросети. Вскрытие монитора пользователем недопустимо ни при каких условиях, вскрытие и обслуживание монитора производится только в специальных мастерских.

8. Все компоненты системного блока получают электроэнергию от блока питания. Правила техники безопасности не запрещают вскрывать системный блок, например, при установке дополнительных внутренних устройств или их модернизации, но это не относится к блоку питания. Блок питания компьютера – источник повышенной пожароопасности, поэтому вскрытию и ремонту он подлежит только в специализированных мастерских. Блок питания имеет встроенный вентилятор и вентиляционные отверстия, поэтому в нем накапливается пыль, которая может вызвать короткое замыкание. Рекомендуется периодически (1-2 раза в год) с помощью пылесоса удалять пыль из блока питания через вентиляционные отверстия без вскрытия системного блока. Особенно важно производить эту операцию перед транспортировкой или наклоном системного блока.

Приложение 2

План воспитательной работы

План воспитательной работы программы «Программирование на Python» корректируется в соответствии с планом воспитательной работы МБОУ «Парабельская СШ им. Н. А. Образцова». Мероприятия в течение учебного года могут варьироваться в зависимости от интересов и пожеланий детей и их законных представителей. В результате реализации воспитательной программы у обучающихся формируется позитивное отношение к окружающему миру, другим людям, самому себе.

Приоритетные направления воспитательной работы на протяжении учебного года и в каникулярный период:

- Гражданско-патриотическое направление.
- Безопасная жизнедеятельность, физическое воспитание, ЗОЖ.
- Самоуправление.
- Интеллектуальное воспитание.
- Эстетическое воспитание.
- Работа с родителями (законными представителями)