УПРАВЛЕНИЕ ОБРАЗОВАНИЯ АДМИНИСТРАЦИИ ГОРОДА ИВАНОВА
МУНИЦИПАЛЬНОЕ БЮДЖЕТНОЕ УЧРЕЖДЕНИЕ
ДОПОЛНИТЕЛЬНОГО ОБРАЗОВАНИЯ
ЦЕНТР ДЕТСКОГО ТВОРЧЕСТВА №4

РЕКОМЕНДОВАНО
Председатель КМС
_____М.И. Безрукова
Протокол № 11
от «15» 07 2022г.

УТВЕРЖДЕНО
Директор МБУ ДО ЦДТ №4
_____ А.Е. Голубев
Приказ № 139-о
от «18» 07 2022г.

Дополнительная общеобразовательная общеразвивающая программа
«Python»
(стартовый уровень)

Направленность: техническая
Профиль: программирование
Возраст детей: 9 - 16 лет
Срок реализации: 1 год

Автор-составитель:
Жаворонкова Татьяна Сергеевна,
педагог дополнительного образования

Иваново, 2022

Раздел №1 «Комплекс основных характеристик программы»

1.1 Пояснительная записка ..... 3 стр.
1.2 Цели и задачи программы ..... 4 стр.
1.3 Содержание программы ..... 5 стр.
1.4 Планируемые результаты ..... 9 стр.

Раздел №2 «Комплекс организационно-педагогических условий»

2.1 Календарный учебный график ..... 9 стр.
2.2 Условия реализации программы ..... 9 стр.
2.3 Формы аттестации обучающихся ..... 9 стр.
2.4 Оценочные материалы ..... 10 стр.
2.5 Методические материалы ..... 10 стр.
2.6 Список литературы ..... 11 стр.
Приложения к программе ..... 12 стр.

Раздел №1. Комплекс основных характеристик программы.

1.1 Пояснительная записка

В настоящее время мы переживаем большие изменения в развитии общества. В современную жизнь человека всё больше внедряются компьютеры и информационные технологии. Всё большее значение приобретает умение человека грамотно обращаться с компьютером, причём зачастую не на пользовательском уровне, а на уровне начинающего программиста. В обязательном школьном курсе информатики программирование нередко представлено лишь на элементарном уровне, на это выделяется недостаточное количество часов. Лишь немногие школы могут себе позволить преподавать программирование на достойном уровне. Следствием этого является формальное восприятие учащимися основ современного программирования и неумение применять полученные знания на практике. Данная дополнительная общеобразовательная общеразвивающая программа создана действующим программистом. В ее содержании большое внимание уделяется практической работе на компьютере, самостоятельному написанию кода.

Актуальность программы вызвана потребностью современного информационного общества в высокообразованных, адаптированных к изменениям специалистах в IT-сфере. Для удовлетворения данной потребности перед дополнительным образованием стоит задача развития человеческого потенциала через выявление талантливых детей, развитие их мотивации и способностей к программированию.

Дополнительная общеобразовательная общеразвивающая программа «Python» является программой **технической направленности.**

Профиль программы – программирование. Программа предполагает работу с языком программирования «Python», который имеет массу преимуществ перед другими языками программирования. «Python» является одним из самых популярных и универсальных языков программирования согласно десяткам рейтингов.

Уровень программы – стартовый.

Новизна Программы заключается в том, что Python дает более широкие возможности в области программирования, чем PascalABC, который входит в школьный курс информатики. На языке Python можно легко и быстро создавать простые компьютерные игры, трёхмерные модели и программировать роботов. Отличие «Python», в котором будут разбираться обучающиеся, от других популярных языков программирования состоит в том, что он является интерпретируемым языком программирования, который не компилируется. Таким образом, до запуска он представляет собой обычный текстовый файл. Соответственно, программировать можно почти на всех платформах, а сам язык логичен и хорошо спроектирован. Кода в нём меньше, чем при использовании других языков программирования, поэтому разработка осуществляется быстрее.

Педагогическая целесообразность программы заключается в привлечении учащихся к занятиям техническим творчеством, что способствует развитию логического мышления, творческих способностей и навыков решения задач программирования. Программирование мотивирует к занятиям в различных научных областях (физики, информатики, алгебры, геометрии и др.), развивает воображение и способствует ранней профориентации подростков. Для достижения поставленных задач занятия проводятся в формате «от простого к сложному». Учащиеся получают знания по основам алгоритмизации и программирования и на их основе, углубляя их, учатся составлять простые и сложные программы.

Адресат программы обучающиеся в возрасте 9–16 лет, мотивированных к обучению

программирования и обладающих системным мышлением. Группы формируются по возрасту: 9-12 лет, 12-16 лет.

Объем и срок освоение программы: программа рассчитана на 1 год (36 учебных недель) при занятиях один раз в неделю по 2 академических часа. Общее количество часов в год – 72.

Форма обучения: дистанционная.

Особенности организации образовательной деятельности: программа рассчитана на постепенное освоение материала, в цельном комплексе занятий. В связи с этим, а также в связи с дистанционным форматом работы, группы являются малокомплектными (6-8 чел.), постоянными. Обучение проводится на встроенной в Python платформе jitsi.

В занятии предусмотрен пятиминутный перерыв для проведения физкультминутки, а также гимнастики для глаз, согласно санитарным правилам и нормам, СанПиН 2.4.3648-20 «Гигиенические требования к видеодисплейным терминалам и персональным электронно-вычислительным машинам и организация работы» и СанПиН 2.4.3648-20 «Санитарно-эпидемиологические правила и нормативы» для учащихся.

1.2 Цель и задачи программы.

Цель программы: создание условий для мотивации обучающихся к программированию посредством языка Python, развитие логического, технического мышления, создание условий для творческой самореализации личности ребёнка.

Задачи:

Предметные:

- изучение конструкций языка программирования Python;
- формирование навыков разработки эффективных алгоритмов и программ на основе изучения языка программирования Python;
- обучение созданию прикладного программного обеспечения;
- расширение кругозора в области программирования.

Метапредметные:

- развитие памяти и внимания, познавательной и творческой активности;
- развитие логического и технического мышления;
- формирование и развитие навыков работы с различными источниками информации, необходимой для решения учебных задач.

Личностные:

- развитие основ коммуникативных отношений внутри проектных групп и в коллективе в целом;
- воспитание этики групповой работы;
- воспитание упорства в достижении результата.

1.3 Содержание программы.
Учебно-тематический план

№ п/п	Название раздела, темы	Количество часов			Форма аттестации/ контроля
		всего	теория	практика	
1	**Раздел 1. Основы языка**				
1.1	Вводное занятие. Начальный мониторинг.	2	1	1	Устный опрос, тестирование
1.2	Введение в Python.	4	2	2	Устный опрос
1.3	Функции ввода-вывода.	4	2	2	Устное тестирование
1.4	Переменные. Численные типы данных.	4	2	2	Тестирование
1.5	Строки. Вложенные конструкции.	4	2	2	Решение минимального набора задач
Раздел 2. Управляющие конструкции					
2.1	Логический тип данных. Условный оператор.	4	2	2	Практическая работа, тест
2.2	Вложенный условный оператор и условный оператор нескольких ветвей.	4	2	2	Решение минимального набора задач
2.3	Цикл while.	2	1	1	Решение минимального набора задач
2.4	Цикл со счётчиком.	2	1	1	Решение минимального набора задач
2.5	Вложенные алгоритмические конструкции	4	2	2	Решение минимального набора задач. Презентация проекта.
Раздел 3. Функции и модули					
3.1	Создание функций.	4	2	2	Устное тестирование
3.2	Локальная и глобальная области видимости.	2	1	1	Устное тестирование
3.3	Использование одной функции внутри другой.	4	2	2	Решение минимального набора задач
3.4	Модули. Использование встроенных модулей стандартной библиотеки.	4	2	2	Решение минимального набора задач
3.5	Создание модулей. Подключение нескольких	6	2	4	Решение минимального

	модулей.				набора задач, практическое задание
Раздел 4. Модуль Turtle. Математика для разработчика.					
4.1	Графический исполнитель Turtle.	4	2	2	Устное тестирование. Практическая работа.
4.2	Turtle. Условный оператор.	4	2	2	Решение минимального набора задач
Раздел 5. Объектно-ориентированное программирование.					
5.1	Объекты, их поля и методы. Обработка событий мыши и клавиатуры.	4	2	2	Устное тестирование, практическая работа
5.2	Классы. Конструктор класса. Наследование	4	2	2	Устный опрос, тестирование, практическая работа
Проектное занятие					
	Обобщение и систематизация пройденного материала.	4	1	3	Проект “Городская среда”. Презентация проекта
	Итого:	**72**	**34**	**38**	

Содержание учебно-тематического плана

Раздел 1. Основы языка

1.1 Вводное занятие.

Начальный мониторинг.

Теория: знакомство с обучающимися, инструктаж.

Практика: начальный мониторинг знаний обучающихся в области программирования.

1.2 Введение в Python.

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

Практика: установка языка программирования Python 3.6 и среды программирования WinglDE 101 6.0. Регистрация на Интернет-ресурсах.

1.3 Функции ввода-вывода.

Теория: изучение базовых правил синтаксиса Python

Практика: отработка на платформе с помощью функции печати данных print() и основных арифметических операторов.

1.4 Переменные. Численные типы данных.
Теория: работа с переменными, отличие типов данных и переход от одного типа к другому с помощью функции int.
Практика: составление арифметических выражений и работы с данными, вводимыми с клавиатуры с помощью функции input(). Новая тема дополняется новым требованием к оформлению кода (требования к именам переменных).

1.5 Строки. Вложенные конструкции.
Теория: методы работы со строками и использование их для решения задач, распознавание и использование команд, применение на практике способы создания и оптимизации качественного кода.
Практика: отработка практико-ориентированных задач.

Раздел 2. Управляющие конструкции
2.1 Логический тип данных.
Теория: Логический тип данных. Логические выражения и операторы. Сложные условные выражения (логические операции and, or, not). Условный оператор. Альтернативное выполнение. Примеры решения задач с условным оператором. Множественное ветвление. Реализация ветвления в языке Python.
Практика: Практическая работа. Логические выражения. Составление программ с ветвлением. Тест. Условные операторы.

2.2 Вложенный условный оператор и условный оператор нескольких ветвей.
Теория: понятие вложенный условный оператор и условная конструкция с несколькими ветвями.
Практика: сравнение разных способов решения задач и применение на практике.

2.3 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.

2.4 Цикл со счётчиком.
Теория: цикл for для повтора действий n-раз и для перебора символов строки.
Практика: решение сюжетных задач.

2.5 Вложенные алгоритмические конструкции
Теория: вложенные алгоритмические конструкции. Цикл в цикл, условный оператор в цикл и др.
Практика: систематизация алгоритмических конструкций, решение заданий, программирование бота. Презентация проектов.

Раздел 3. Функции и модули
3.1 Создание функций.
Теория: понимание функции как фрагмента кода программы (подпрограммы), команды которого выполняются при обращении по имени.

Практика: работа со встроенными функциями и создание собственных используя операторы def и return.

3.2 Локальная и глобальная области видимости.
Теория: область видимости переменных (scope) в Python, глобальные и локальные переменные, сокрытие значений, константы, global и nonlocal.
Практика: решение сюжетных задач

3.3 Использование одной функции внутри другой.
Теория: вложенные функции. Определение внутренней функции. Инкапсуляция. Замыкания.
Практика: работа со вложенными функциями. Решение практических задач.

3.4 Модули. Использование встроенных модулей стандартной библиотеки.
Теория: обзор стандартной библиотеки. Возможности модулей стандартной библиотеки Python: random и time.

Практика: решение сюжетных задач

3.5 Создание модулей. Подключение нескольких модулей.
Теория: создание собственных модулей. Подключение модулей.
Практика: решение задач. Программирование программы – автоответчик.

Раздел 4. Модуль Turtle. Математика для разработчика.
4.1 Графический исполнитель Turtle.
Теория: возможности модуля turtle для отрисовки графических примитивов. Математика перемещения исполнителя.
Практика: рисование простейших геометрических фигур с помощью команд модуля turtle. Система координат, координаты точки и как переместить исполнителя в нужную точку на экране.

4.2 Turtle. Условный оператор.
Теория: возможности модуля turtle для отрисовки графических примитивов.
Практика: изучение новых команд модуля turtle и возможности их использования в циклических конструкциях. Решение сюжетных задач.

Раздел 5. Объектно-ориентированное программирование.
5.1 Объекты, их поля и методы. Обработка событий мыши и клавиатуры.
Теория: модель «черного ящика». Понятия класса и объекта. Интерфейс класса: поле и метод. Принципы ООП: инкапсуляция, наследование, полиморфизм
Практика: решение сюжетных задач.

5.2 Классы. Конструктор класса. Наследование
Теория: создание и использование класса. Инициализация класса. Атрибуты объекта и атрибуты класса. Проектирование иерархии.
Практика: практическая работа. Проектирование иерархии.

6.1 Проектное занятие.
Теория: обобщение и систематизация пройденного материала.
Практика: проект “Городская среда”. Презентация проекта.

1.4 Планируемые результаты.

Предметные:

- знает конструкций языка программирования Python;
- сформированы навыки разработки эффективных алгоритмов и программ на основе изучения языка программирования Python;
- умеет создавать простейшие программы;

Метапредметные:

- сформированы навыки развития памяти и внимания, познавательной и творческой активности;
- развивается логическое и техническое мышление;
- сформированы и развиты навыки работы с различными источниками информации, необходимой для решения учебных задач.

Личностные:

- развиты основы коммуникативных отношений внутри проектных групп и в коллективе в целом;
- развиты навыки групповой работы;
- воспитано упорство в достижении результата.

Раздел №2. Комплекс организационно-педагогических условий.

2.1. Календарный учебный график

Количество учебных недель – 36;
Даты начала и окончания обучения: учебный год начинается с 1 сентября, окончание учебного года 31 мая. (Приложение №1)

2.2 Условия реализации программы

Дополнительная общеобразовательная общеразвивающая программа «Python» реализуется с учетом требований Постановления Главного государственного санитарного врача РФ №2 от 28.01.2021 г. «Об утверждении санитарных правил и норм СанПиН 1.2.3685-21 «Гигиенические нормативы и требования к обеспечению безопасности и (или) безвредности для человека факторов среды обитания», распоряжения Правительства Российской Федерации от 02.12.2021 г. № 3427-р «Об утверждении стратегического направления в области цифровой трансформации образования, относящейся к сфере деятельности Министерства просвещения Российской Федерации» программа может быть адаптирована к дистанционному формату».

Обучение проводится на встроенной в Python платформе jitsi.

Программное обеспечение:

операционная система – любая, желательно Windows;
свободно распространяемое ПО IDLE и PyCharm

Информационное обеспечение:

доступ в интернет, Microsoft Teams

Материально-техническое обеспечение:

Оборудование:

Десктоп (Персональный компьютер) / Лэптоп (Ноутбук): Центральный процессор - ЦП с архитектурой x64 не старше 5 лет Intel серии Core - Core i3, 5, 7, 9 - не старше третьего

поколения; серии Pentium и Celeron не рекомендуются. AMD серии FX - не ниже FX-4350, все модели серии Ryzen. старшие модели серий A10 и A12 (9700, 9800), Athlon 200GE. Дискретная видеокарта с поддержкой DirectX 10, Shader Model 2.0 и новее; ОЗУ: DDR3 и новее, объемом от 4 Гб. Рекомендуется 8 Гб и больше.
Клавиатура, мышь, звуковая карта, микрофон, веб-камера (не обязательно, по желанию).

2.3 Формы аттестации

Система контроля знаний и умений обучающихся представляется в виде учёта результатов по итогам выполнения заданий отдельных кейсов и посредством наблюдения, отслеживания динамики развития. Для обучающихся по программе формами контроля являются:

- наблюдение
- выполнение групповых и индивидуальных практических заданий
- тесты
- рефлексия
- презентация проектов

2.4 Оценочные материалы

Результативность обучения обеспечивается применением различных форм, методов и приемов, которые тесно связаны между собой и дополняют друг друга. Большая часть занятий отводится практической работе, в ходе практической деятельности педагог тактично контролирует, советует, направляет обучающихся. Для определения уровня знаний и умений обучающихся проводятся мониторинги: в начале года - входной мониторинг, в середине года - промежуточный мониторинг, в конце года — итоговый мониторинг. Основной способ оценки – самооценка выполнения программ на языке программирования. Одной из форм диагностики результатов является зачет (тесты) или выполнение практических задач, презентация проектов. Текущий контроль уровня усвоения материала осуществляется по окончании изучения каждой темы – выполнением практических заданий, тестов. Итоговый контроль проходит в конце учебного года – презентация проектов. (Приложение №2).

2.5 Методические материалы

Образовательный процесс осуществляется в дистанционной форме.

В образовательном процессе используются следующие методы:

- словесные (беседа, опрос, дискуссия и т.д.);
- метод проблемного изложения (постановка проблемы и решение её самостоятельно или группой);
- метод проектов;
- наглядные: демонстрация схем, таблиц, диаграмм;
- практические: практические задания;
- анализ и решение проблемных ситуаций и т.д.

2.6 Список литературы

Нормативные документы:

1. Федеральный закон «Об образовании в Российской Федерации» от 29.12.2012 N 273-ФЗ.

2. Стратегия развития воспитания в Российской Федерации на период до 2025 года. Распоряжение Правительства Российской Федерации от 29 мая 2015 г. № 996-р.

3. Письмо Министерства образования и науки РФ от 18.11.2015г. № 09- 3242. «О направлении Методических рекомендаций по проектированию дополнительных общеразвивающих программ (включая разноуровневые)».

4. Приказ Министерства просвещения России от 09.11.2018 г. № 196 «Об утверждении Порядка организации и осуществления образовательной деятельности по дополнительным общеобразовательным программам».

5. Приказ Министерства образования и науки РФ от 23.08.2017 г. № 816 «Порядок применения организациями, осуществляющими образовательную деятельность, электронного обучения, дистанционных образовательных технологий при реализации образовательных программ».

6. Постановление Главного санитарного врача РФ от 28.09.2020 № 28 «Об утверждении санитарных правил СП 2.4.3648-20 «Санитарно-эпидемиологические требования к организациям воспитания и обучения, отдыха и оздоровления детей и молодежи».

7. Постановление Главного государственного санитарного врача РФ №2 от 28.01.2021 г. «Об утверждении санитарных правил и норм СанПиН 1.2.3685-21 «Гигиенические нормативы и требования к обеспечению безопасности и (или) безвредности для человека факторов среды обитания».

8. Распоряжение Правительства Российской Федерации от 02.12.2021 г. № 3427-р «Об утверждении стратегического направления в области цифровой трансформации образования, относящейся к сфере деятельности Министерства просвещения Российской Федерации» программа может быть адаптирована к дистанционному формату».

Список литературы, использованной при написании программы:

1. Задачи по программированию. Под ред. С. М. Окулова, М.: БИНОМ. Лаборатория знаний, 2006.

2. Информатика и ИКТ. Задачник-практикум в 2 частях. Под ред. И. Г. Семакина и Е. К. Хеннера. М.: БИНОМ. Лаборатория знаний, 2014.

3. Лутц М. Изучаем Python. СПб.: Символ-Плюс, 2011. 26

4. Окулов С. М. Основы программирования. М.: Бином. Лаборатория знаний, 2012.

5. Поляков К. Ю., Еремин Е. А. Информатика. Углублённый уровень. Учебник для 10 класса в 2 частях. М.: БИНОМ. Лаборатория знаний, 2014.

6. Эльконин, Д. Б. Детская психология: учеб. пособие для студ. высш. учеб. заведений / Д. Б. Эльконин; ред.сост. Б. Д. Эльконин. – 4-е изд., стер. – М.: Издательский центр «Академия», 2007. – 384 с

7. Python для детей. Самоучитель по программированию / Джейсон Бриггс ; пер. с англ. Станислава Ломакина ; [науч. ред. Д. Абрамова]. — М.: Манн, Иванов и Фербер, 2017 — 320 с.

8. К. Вордерман и др. Программирование для детей: Иллюстрированное руководство по языкам

Scratch и Python, 224 стр. Издательство: Манн, Иванов и Фербер, 2017 г.

9. Программирование для детей на языке Python, 96 стр. Издательство: АСТ, 2017 г.

10. Б. Пэйн. Python для детей и родителей, 352 стр. Издательство: Эксмо, 2017 г.

11. П. Томашевский. Привет, Python! Моя первая книга по программированию, 256 стр. Издательство: Наука и Техника, 2018 г.

Электронные ресурсы:

1. Материалы и презентации к урокам в LMS Яндекс. Лицея.

2. Сайт «Python 3 для начинающих» – pythonworld.ru.

3. Сайт «Питонтьютор» – pythontutor.ru.

Приложение №1

Календарный учебный график

№	Дата (число, месяц)	Форма занятий	Кол-во часов	Раздел УТП	Тема занятий
1	15.09	дистанционная	2	Раздел 1. Основы языка	Вводное занятие. Начальный мониторинг.
2	22.09	дистанционная	2	Раздел 1. Основы языка	Введение в Python.
3	29.09	дистанционная	2	Раздел 1. Основы языка	Введение в Python.
4	6.10	дистанционная	2	Раздел 1. Основы языка	Функции ввода-вывода.
5	13.10	дистанционная	2	Раздел 1. Основы языка	Функции ввода-вывода.
6	20.10	дистанционная	2	Раздел 1. Основы языка	Переменные. Численные типы данных.
7	27.10	дистанционная	2	Раздел 1. Основы языка	Переменные. Численные типы данных.
8	3.11	дистанционная	2	Раздел 1. Основы языка	Строки. Вложенные конструкции.
9	10.11	дистанционная	2	Раздел 1. Основы языка	Строки. Вложенные конструкции.
10	17.11	дистанционная	2	Раздел 2. Управляющие конструкции	Логический тип данных. Условный оператор.
11	24.11	дистанционная	2	Раздел 2. Управляющие конструкции	Логический тип данных. Условный оператор.
12	1.12	дистанционная	2	Раздел 2. Управляющие конструкции	Вложенный условный оператор и условный оператор нескольких ветвей.
13	8.12	дистанционная	2	Раздел 2. Управляющие конструкции	Вложенный условный оператор и условный оператор нескольких ветвей.
14	15.12	дистанционная	2	Раздел 2. Управляющие конструкции	Цикл while.
15	22.12	дистанционная	2	Раздел 2. Управляющие конструкции	Цикл со счётчиком.
16	29.12	дистанционная	2	Раздел 2. Управляющие конструкции	Вложенные алгоритмические конструкции
17	12.01	дистанционная	2	Раздел 2. Управляющие конструкции	Вложенные алгоритмические конструкции

18	19.01	дистанционная	2	Раздел 3. Функции и модули	Создание функций.
19	26.01	дистанционная	2	Раздел 3. Функции и модули	Создание функций.
20	2.02	дистанционная	2	Раздел 3. Функции и модули	Локальная и глобальная области видимости.
21	9.02	дистанционная	2	Раздел 3. Функции и модули	Использование одной функции внутри другой.
22	16.02	дистанционная	2	Раздел 3. Функции и модули	Использование одной функции внутри другой.
23	2.03	дистанционная	2	Раздел 3. Функции и модули	Модули. Использование встроенных модулей стандартной библиотеки.
24	9.03	дистанционная	2	Раздел 3. Функции и модули	Модули. Использование встроенных модулей стандартной библиотеки.
25	16.03	дистанционная	2	Раздел 3. Функции и модули	Создание модулей. Подключение нескольких модулей.
26	23.03	дистанционная	2	Раздел 3. Функции и модули	Создание модулей. Подключение нескольких модулей.
27	30.03	дистанционная	2	Раздел 4. Модуль Turtle. Математика для разработчика.	Графический исполнитель Turtle.
28	6.04	дистанционная	2	Раздел 4. Модуль Turtle. Математика для разработчика.	Графический исполнитель Turtle.
29	13.04	дистанционная	2	Раздел 4. Модуль Turtle. Математика для разработчика.	Turtle. Условный оператор.
30	20.04	дистанционная	2	Раздел 4. Модуль Turtle. Математика для разработчика.	Turtle. Условный оператор.
31	27.04	дистанционная	2	Раздел 5. Объектно-ориентированное программирование.	Объекты, их поля и методы. Обработка событий мыши и клавиатуры.
32	4.05	дистанционная	2	Раздел 5. Объектно-ориентированное программирование.	Объекты, их поля и методы. Обработка событий мыши и клавиатуры.
33	11.05	дистанционная	2	Раздел 5. Объектно-ориентированное программирование.	Классы. Конструктор класса. Наследование
34	18.05	дистанционная	2	Раздел 5. Объектно-ориентированное программирование.	Классы. Конструктор класса. Наследование
35	25.05	дистанционная	2	Проектное занятие	Обобщение и систематизация пройденного материала.

36	1.06	дистанционная	2	Проектное занятие	Обобщение и систематизация пройденного материала.

Приложение №2

Оценочные материалы.

Тест.

Задание: добавьте каждое предложение по смыслу.

1. Программа Python называется…
2. Расширение файла Python - as….
3. Переменная в Python - это…
4. Регистр букв в идентификаторах значение…
5. Выражение в Python - это…
6. Символ # в Python обозначает…
7. … в Python это тип данных для вещественных чисел, встроенный в Python по умолчанию.
8. Операция 3**4 - это…
9. 345 - …тип данных.
10. Операция 46%10 – это…
11. Функция round (d) – это…
12. Функция input () – предназначена для…
13. Для вывода данных есть функция в Python - …
14. …в Python это логический тип данных, встроенный в Python по умолчанию.
15. Строки – это…
16. A=’pri’, s=’vet’. A+s – это…
17. E=’no’. E*5 – это…
18. К элементу в строке можно обратится по …
19. s=’asdfgh’ print (s [-1]). Программа выведет…
20. s=’asdfgh’ print (s [2:4]). Программа выведет…
21. Функция len (строка) – возвращает…
22. Списки – это…
23. Пример списка - …
24. Словари – это…
25. Пример словаря -…
26. Условный оператор в Python - …
27. Цикл for называется циклом…

Критерии оценки проекта.

Оценка работ проводится с учетом критериев:

- Актуальность проекта и практическая применимость (от 0 до 2 баллов). Критерий оценивает соответствие темы проекта прикладным проблемам, перспективы возможного практического использования результатов проекта.
- Новизна и сложность проекта (от 0 до 3 баллов). Критерий оценивает новизну выбранного в проекте подхода (расширенная постановка лабораторных работ; самостоятельно изученные задачи, алгоритмы и технологии), сложность используемых алгоритмов и т.п.
- Сложность программной разработки (от 0 до 6 баллов). Критерий оценивает сложность выполненной программной разработки (используемые технологии, программная архитектура, объем программного кода, использование внешних библиотек, уровень выполненного тестирования).
- Качество пользовательского интерфейса (от 0 до 2 баллов). Критерий оценивает дружественность пользовательского интерфейса (привлекательность, доступность,

простота использования и т.п.) и наличие наглядной демонстрации принципов работы реализованных алгоритмов.

- Уровень проектного выступления (от 0 до 4 баллов). Критерий оценивает качество подготовленной проектной презентации и уровень выступления с представлением проекта (соблюдение регламента, понятность изложения основных результатов проекта, ответы на вопросы и т.п.)