



Методические рекомендации по теме

«Циклы с неограниченным количеством повторений»

Цель:

- понимание структуры и принципов работы цикла с неограниченным количеством повторений на языке Python.

Задачи:

- расширение кругозора обучающихся в области информатики и программирования;
- знакомство с основами работы с циклами с неограниченным количеством повторений на языке Python;
- решение программных прикладных задач с циклами с неограниченным количеством повторений на языке Python;
- ранняя профориентация школьников, профессиональная деятельность программиста;
- развитие интеллектуальных способностей, логического и критического мышления.

Планируемые результаты

Личностные: обучающиеся получают навыки активной коммуникации в группе, осознанной ориентировки в мире ИТ профессий, постановки собственных образовательных задач и владение первичными навыками деятельностного анализа и критической оценки получаемой информации.

Предметные: обучающиеся получают представления: о циклах с неограниченным числом повторений в языке Python; об основных операциях с циклами с неограниченным количеством повторений на языке программирования «Python»; о прикладном использовании операций с циклами с неограниченным количеством повторений в программных проектах; о возможностях и особенностях применения циклов с неограниченным количеством повторений в практике работы программиста.

Метапредметные: обучающиеся получают возможность владения обще предметными понятием «цикл», «повторение», «бесконечный цикл»; владение информационно-логическими умениями; владение умениями самостоятельно планировать пути достижения целей; умениями принятия решений и осуществления осознанного выбора; повысят уровень ИКТ-компетентности.

Материалы к занятию

Приложение 1: Сценарный план видеоролика

Приложение 2: Домашние задание и практика

Приложение 3: Краткие организационно-методические рекомендации по организации работы на занятии

Ход проведения урока

1. Организационный момент.

Мотивация на учебную деятельность.

Приветствие учащихся, сообщение темы и целей занятия (мы уже знаем, про циклы с заданным числом повторений (итераций) - «конечными циклами», теперь познакомимся с «бесконечным циклом»; научимся использовать бесконечные циклы для разработки программ).

Проблемная дискуссия по вопросам:

- Что такое цикл с неограниченным числом повторений. Приведите примеры из жизни?
- Для чего может пригодиться цикл с неограниченным числом повторений при программировании в Python?
- Какой синтаксис вы бы использовали при работе с такими циклами? Попробуйте придумать свою запись.

Итоги дискуссии (обобщаются преподавателем и фиксируются ответы учеников на доске, чтобы вернуться к ним и оценить правильность предположений учеников на этапе рефлексии):

- бесконечный цикл позволяет решать задачи, в которых количество повторов заранее неизвестно;
- также бесконечный цикл используется в программах, где условие для завершения цикла не является простым.

Преподаватель называет ученикам тему и цели урока.

2. Вводный блок.

Тема.

Преподаватель при необходимости останавливая трансляцию, комментируя дополнительно тему занятия.

**см. сцены 1 – 2 (здесь и далее приводится Таблица «Содержание видеоролика». Приложение 1).*

3. Блок повторения.

Блиц-опрос.

Преподаватель предлагает ученикам ответить на **5 вопросов** по предыдущей теме; задания выполняются в сопровождении видеоролика с использованием таймера; ученики выполняют задания, голосуют, обсуждают результаты. Процедура голосования определяется инструкцией **в сцене 3**; учитель должен убедиться, что всем понятна процедура голосования. *Преподаватель может поставить ролик на паузу и обсудить результаты голосования; объяснить правильный ответ руководствуясь материалами предыдущего занятия*

**см. сцены 3 – 7*

4. Теоретический блок.

Цикл с неограниченным числом повторений.

Продолжение демонстрации ролика с дальнейшим обсуждением вопросов:

- Какие команды используются при работе с бесконечными циклами? Что они значат?
- Как связаны бесконечный цикл и оператор **while**?
- Каким образом можно остановить выполнение бесконечного цикла?

При необходимости преподаватель может поставить ролик на паузу и дать дополнительные пояснения по материалу; если ответы на вопросы вызывают у учеников затруднения, преподаватель может вывести нужную сцену ролика на экран для помощи ученикам.

**см. сцена 8 – 10*

5. Блок заданий.

Проекты: «Удвоение до тысячи», «Загадка».

К началу демонстрации блока заданий ученики должны занять рабочие места и запустить Python (терминал IDLE) на своих компьютерах. Блок включает **2 практических задания** для учеников с последующим разбором. Задания представляют собой 2 небольших программных проекта с использованием «бесконечных циклов».

После выполнения заданий 1 и 2 ученики получают два работающих программных продукта – программа для поэтапного удвоения значения любого числа в пределах 1000 и программа для решения загадок.

На сцене разбора задания преподаватель ставит ролик на паузу и вместе с учениками проводит разбор задания.

**см. сцены 9 – 19*



6. Рефлексия. Сообщение домашнего задания.

Завершаем демонстрацией ролика и кратким обобщением материалов занятия. Преподаватель возвращается к зафиксированному в ходе дискуссии в начале урока предположениям учеников и обсуждает насколько их предположения были правильными, делаются выводы.

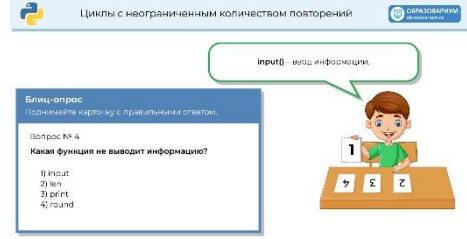
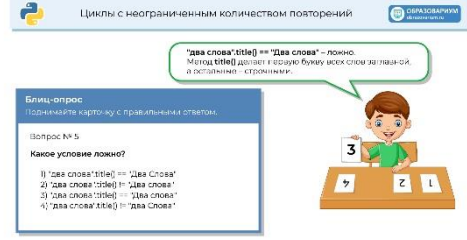
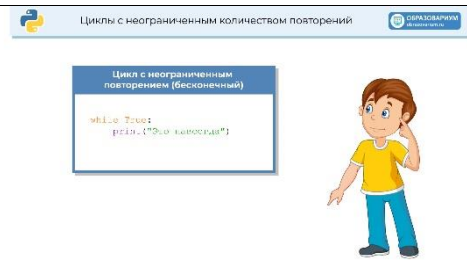
Преподаватель дает ученикам домашнее задание к следующему занятию (*Приложение 2*). **см. сцена 20*

В таблице «Содержание видеоролика» представлен краткий текст из видеоролика, примеры заданий и задач, которые будут демонстрироваться на экране. Учитель при подготовке к уроку может ознакомиться с содержанием видеоролика в текстовом формате, при необходимости распечатать фрагменты текста или примеры заданий и задач для использования в работе с учениками. Распечатанные тексты и задания из таблицы также можно применять в качестве раздаточного материала как на уроке, так и для домашних заданий.

Таблица. Содержание видеоролика

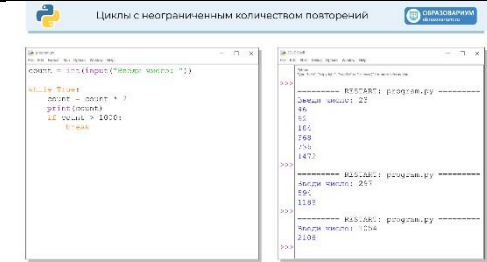
Название блока	Содержание блока и комментарии	Фрагменты из видеоролика	№ сцен
Вводный блок. Мы узнаем	Обозначаем ученикам тему и цели урока. Циклы с неограниченным количеством повторений.	 Сцена 1	1 2
	Циклы с условиями и циклы, имеющие определенное количество повторов, дают возможность создавать программы, не прибегая к многочисленным повторам в написании кода. Теперь познакомимся с ещё одним инструментом этого раздела – бесконечным циклом. Его также называют цикл, с неограниченным количеством повторений	 Сцена 2	

<p>Блок повторения.</p> <p>Блиц-опрос</p>	<p><i>Повторение материала предыдущего урока; на столе имеются пронумерованные карточки; после каждого вопроса выбираем ту, номер которой, совпадает с правильным ответом.</i></p> <p>Первый вопрос. Поднимите карточку, какое последнее значение будет у счетчика?</p> <p>for i in range (0,7,4)</p> <p><i>Ответ 4. Счетчик получит значения 0 и 4. Последним значением будет 4.</i></p>	<p>Циклы с неограниченным количеством повторений</p> <p>Блиц-опрос Поднимайте карточку с правильным ответом.</p> <p>Вопрос №1 какое последнее значение будет у счетчика? for i in range (0, 7, 4)</p> <p>Поднимите карточку с соответствующим числом.</p> <p>Счетчик получит значения 0 и 4. Последним значением будет 4.</p> <p>4</p> <p>5 2 1</p> <p>Сцена 3</p>	<p>3</p> <p>4</p> <p>5</p> <p>6</p> <p>7</p>
	<p>Второй вопрос. Поднимите карточку, какое последнее значение будет у счетчика?</p> <p>for i in range (5,1, -2)</p> <p><i>Ответ 3. Счетчик получит значения 5 и 3. Последним значением будет 3.</i></p>	<p>Циклы с неограниченным количеством повторений</p> <p>Блиц-опрос Поднимайте карточку с правильным ответом.</p> <p>Вопрос №2 какое последнее значение будет у счетчика? for i in range (5, 1, -2)</p> <p>Поднимите карточку с соответствующим числом.</p> <p>Счетчик получит значения 5 и 3. Последним значением будет 3.</p> <p>3</p> <p>4 2 1</p> <p>Сцена 4</p>	
	<p>Третий вопрос. Выберите правильный вариант среза "лоб" из слова "килобайт"</p> <p>1) [2: 4] 2) [3: 6] 3) [2: 5] 4) [3: 5]</p> <p><i>Ответ 3. "лоб" == "килобайт"[2: 5]</i></p>	<p>Циклы с неограниченным количеством повторений</p> <p>Блиц-опрос Поднимайте карточку с правильным ответом.</p> <p>Вопрос №3 Выберите правильный вариант среза "лоб" из слова "килобайт".</p> <p>1) [2: 4] 2) [3: 6] 3) [2: 5] 4) [3: 5]</p> <p>"килобайт" 0 1 2 3 4 5 6 7</p> <p>"лоб" == "килобайт"[2: 5]</p> <p>3</p> <p>4 2 1</p> <p>Сцена 5</p>	

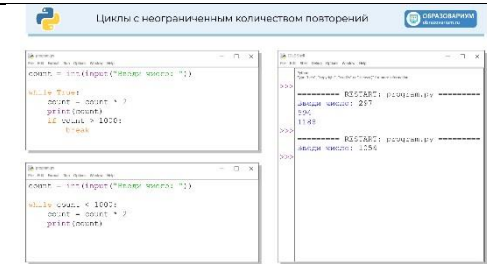
	<p>Четвертый вопрос. Какая функция не выводит информацию?</p> <ol style="list-style-type: none"> 1) input 2) len 3) print 4) round <p><i>Ответ 1. input() – ввод информации.</i></p>	 <p>Циклы с неограниченным количеством повторений</p> <p>Блиц-опрос Поднимайте картушку с правильными ответом.</p> <p>Вопрос № 4 Какая функция не выводит информацию?</p> <ol style="list-style-type: none"> 1) input 2) len 3) print 4) round <p>input() – ввод информации.</p> <p>Сцена 6</p>	
	<p>Пятый вопрос. Какое условие ложно?</p> <ol style="list-style-type: none"> 1) "два слова".title() == "Два Слова" 2) "два слова".title() != "Два слова" 3) "два слова".title() == "Два слова" 4) "два слова".title() != "два Слова" <p><i>Ответ 3. "два слова".title() == "Два слова" – ложно. Метод title() делает первую букву всех слов заглавной, а остальные – строчными.</i></p>	 <p>Циклы с неограниченным количеством повторений</p> <p>Блиц-опрос Поднимайте картушку с правильными ответом.</p> <p>Вопрос № 5 Какое условие ложно?</p> <ol style="list-style-type: none"> 1) "два слова".title() == "Два Слова" 2) "два слова".title() != "Два слова" 3) "два слова".title() == "Два слова" 4) "два слова".title() != "два Слова" <p>"два слова".title() == "Два слова" – ложно. Метод title() делает первую букву всех слов заглавной, а остальные – строчными.</p> <p>Сцена 7</p>	
<p>Теоретический блок.</p> <p>Бесконечный цикл.</p>	<p><i>При необходимости преподаватель может поставить ролик на паузу и дать дополнительные пояснения по материалу</i></p> <p>Возникает вопрос – зачем нужна такая программа? Это получается, запустил ее и больше никогда в жизни к этому компьютеру не подойдешь?</p>	 <p>Циклы с неограниченным количеством повторений</p> <p>Цикл с неограниченным повторением (бесконечный)</p> <pre>while True: print("Эта программа")</pre> <p>Сцена 8</p>	<p>8 9 10</p>

	<p>Бесконечный цикл можно остановить командой break, которая срабатывает после некоего условия.</p> <p>Поэтому можно сказать, что бесконечный цикл – это разновидность цикла с условием. У них даже общий оператор while. Но зачем это нужно?</p> <p>Чтобы организовать цикл с условием нам сразу необходимы используемые данные. И само условие, как правило, очень конкретно.</p>	<div data-bbox="1503 231 1962 486"> </div> <p>Сцена 9</p>	
	<p>Вот пример: нам сначала нужна переменная num и условие выхода из цикла – num не равно нулю.</p> <p>Как можно сделать это при помощи бесконечного цикла. Обратите внимание: сначала мы вошли в цикл, и только потом создали переменную num.</p> <p>Код стал длиннее на одну строку, но иногда – особенно если условий для прекращения цикла несколько – это все равно удобнее.</p>	<div data-bbox="1503 630 1962 845"> </div> <p>Сцена 10</p>	
<p>лок заданий.</p> <p>Практические задания:</p> <p>Задание 1</p>	<p><i>После окончания дикторского текста запускается таймер на 4 мин.</i></p> <p>Задание 1. Проект «Удвоение до 1000»</p> <ul style="list-style-type: none"> – Запросить число и занести ответ в переменную count – Организовать бесконечный цикл <ul style="list-style-type: none"> ◦ Умножить count на 2 ◦ Вывести count на экран ◦ Если count > 1000 break 	<div data-bbox="1503 1029 1962 1300"> </div> <p>Сцена 11</p>	<p>11</p> <p>12</p> <p>13</p> <p>14</p> <p>15</p> <p>16</p> <p>17</p> <p>18</p> <p>19</p>

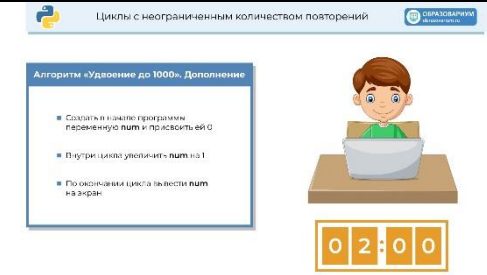
<p>Разбор задания 1. Наш код может выглядеть так.</p> <pre>count = int(input("Введи число: ")) while True: count = count * 2 print(count) if count > 1000: break</pre> <p><i>После окончания времени провести разбор решения, анализируя код. Обратите внимание – даже если мы введем число больше тысячи – одно умножение все-таки состоится.</i></p>	<p>Еще одно отличие бесконечного цикла от цикла с условием. Бесконечный цикл всегда будет запущен, в отличие от условного, который может и ни разу не сработать.</p>	<p><i>После окончания дикторского текста запускается таймер на 2 мин.</i></p> <p>Задание 1. Проект «Удвоение до 1000» (дополнение)</p> <ul style="list-style-type: none"> – Создать в начале программы переменную num и присвоить ей 0 – Внутри цикла увеличить num на 1 – По окончании цикла вывести num на экран
--	--	---



Сцена 12



Сцена 13



Сцена 14

Разбор задания 1 (дополнение).

Наш код может выглядеть так.

```
num = 0
```

```
count = int(input("Введи число: "))
```

```
while True:
```

```
    num = num + 1
```

```
    count = count * 2
```

```
    print(count)
```

```
    if count > 1000:
```

```
        break
```

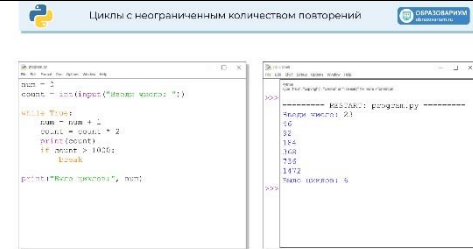
```
print("Было циклов:", num)
```

В целом этот проект не показал большой разницы между условным и бесконечным циклами. Все потому, что условие прекращения было простое и никаких дополнительных действий при этом не происходило.

После окончания дикторского текста запускается таймер на 6 мин.

Задание 2. Проект «Загадка»

- Вывести на экран текст любой загадки
- Организовать бесконечный цикл
- Спросить ответ и занести его в переменную `answer`
- Если `answer ==` верному ответу
 - напечатать «молодец»
 - остановить цикл
- Если `answer ==` «сдаюсь»
 - напечатать ответ
 - остановить цикл



Сцена 15



Сцена 16

Разбор задания 2.

Код проекта может быть таким.

```
print("Что такое: зимой звезда, весной вода?")
```

```
while True:
```

```
    answer = input("Твой ответ: ")
```

```
    if answer == "снег":
```

```
        print("Верно")
```

```
        break
```

```
    elif answer == "сдаюсь":
```

```
        print("Это снег")
```

```
        break
```

Здесь присутствуют два условия для окончания цикла, и каждое сопровождается выводом пояснительной информации. Создать подобный проект с использованием условного цикла было бы проблематично.

После окончания дикторского текста запускается таймер на 1 мин.

Задание 2. Проект «Загадка» (продолжение)

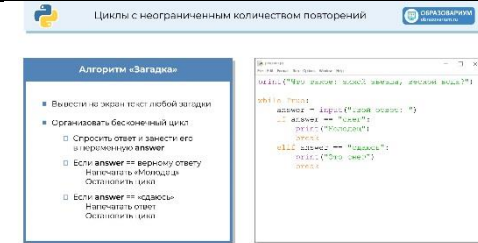
- Вставить в проект метод изменения регистра букв на строчный.

Разбор задания 2 (продолжение).

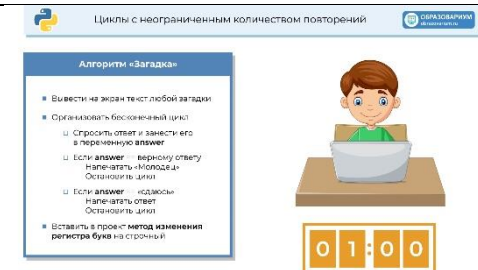
Необходимо при вводе ответа дописать метод **lower**.

```
answer = input("Твой ответ: ").lower()
```

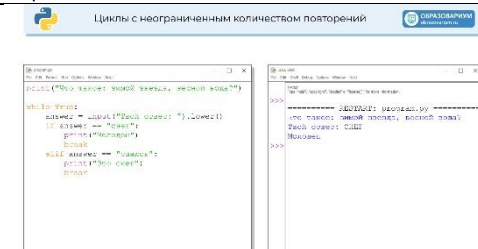
После этого, программа будет принимать текстовую информацию, не обращая внимания на регистр.




Сцена 17



Сцена 18



Сцена 19

Блок завершения занятия. Рефлексия. Сообщение домашнего задания	<i>Завершаем демонстрацией ролика и кратким обобщением материалов занятия.</i> Подведем итоги. – Умеем применять: <ul style="list-style-type: none"> • цикл с условием • конечный цикл • бесконечный цикл <i>Преподаватель дает ученикам домашнее задание к следующему занятию (Приложение 2).</i>	 Сцена 20	20
---	--	--	----

Приложение 2

Домашнее задание

Придумайте примеры бесконечных циклов в окружающем мире и условия, способные их прервать.

Практика

Проект «Новый максимум»

В бесконечном цикле запрашивайте у пользователя целое положительное число. Если число превосходит максимальное, выведите соответствующее сообщение на экран и замените максимальное значение на введенное.

Когда будет введено отрицательное число, выведите соответствующее сообщение и остановите работу программы.

Проект «Четные и нечетные»

В бесконечном цикле запрашивайте у пользователя целое число. Определите, является оно четным или нет, выведите соответствующее сообщение на экран. Ведите подсчет количества введенных четных и нечетных чисел.

Когда будет введен 0, остановите работу программы и выведете, сколько было введено четных и сколько нечетных чисел.

Краткие организационно-методические рекомендации по организации работы на занятии

«Циклы с неограниченным количеством повторений».

В начале занятия можно рассмотреть домашнее задание – пример собственного алгоритма с использованием конечного цикла. Заодно повторить как создается конечный цикл, какими командами, какие есть варианты цикла (восходящий, нисходящий) и как это влияет на параметры, а также чем он отличается от цикла с условием.

Все повторения лучше сопровождать примерами на доске.

Перед просмотром блока повторения из ролика необходимо раздать дидактический материал для выполнения заданий из блока повторение (по 4 пронумерованных карточки)

Во время голосований карточками можно останавливать ролик и вести учет правильных ответов. По окончании блока – отметить тех, у кого наилучший результат.

Далее карточки необходимо собрать.

После теоретического материала необходимо удостовериться, что ребята поняли следующее: бесконечный цикл — это разновидность цикла с условием. А какой из них надо использовать в конкретной ситуации – зависит от сложности условий.

После создания и тестирования первого проекта, можно заменить умножение на некую более сложную формулу и (как вариант) ввести не одно, а несколько чисел для ее работы.