



## Методические рекомендации по теме

«Переменная и оператор присваивания. Типы переменных и операции с ними»

### Цель:

- объяснение понятия тип данных, знакомство с различными типами данных.

### Задачи:

- расширение кругозора обучающихся в области информатики и программирования;
- знакомство с понятием данные и различными типами данных в программировании;
- 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 целей и задач.

*Предметные:* обучающиеся получают представления о понятии «данные» в программировании и различных типах данных; о базовых операциях с различными типами данных и понятии «переменная»; изучат основные действия и использование переменных.

*Метапредметные:* обучающиеся получают возможность овладеть общепредметными понятиями «данные», «переменная» и информационно-логическими умениями; умениями самостоятельно планировать пути достижения целей и принимать решения в осуществлении осознанного выбора; повысить уровень ИКТ – компетентности.

## **Материалы к занятию**

Приложение 1: Сценарный план ролика.

Приложение 2: Домашнее задание и практика.

Приложение 3: Краткие организационно-методические рекомендации.

## **Ход проведения урока**

### **1. Организационный момент.**

#### **Мотивация на учебную деятельность.**

Приветствие учащихся, сообщение темы и целей занятия (мы узнаем, что такое «данные» в программировании и каких типов они бывают; познакомимся с «переменными» и научимся выполнять с ними основные операции).

#### ***Проблемная дискуссия*** по вопросам:

- Что такое данные и как информация становится данными?
- Как можно определить это понятие?
- С какими данными мы встречаемся в жизни?
- Можно ли разделить данные по типам?
- Что такое в вашем понимании «биометрические данные», «персональные данные», «база данных»?
- Что вы представляете, когда слышите слово «переменная»?

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

- любая закодированная информация – это данные (когда мы записали, зарисовали, оцифровали и зафиксировали информацию как-то иначе она становится данными); данные можно разделить по типу (текстовые, числовые, графические).

Преподаватель называет ученикам тему и цели урока.

## **2. Вводный блок.**

### **Мы узнаем.**

Преподаватель предлагает ученикам просмотреть ролик (при необходимости останавливая трансляцию и комментируя дополнительно)

*\*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 4; 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.**

### **Данные и переменные.**

Продолжение демонстрации ролика с дальнейшим обсуждением вопросов:

- Что такое переменная?
- Какая информация может «храниться» в переменной?

- Какие правила нужно соблюдать при создании переменных?
- Какие типы переменных существуют?
- Может ли переменная в Python менять свой тип?

*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. сцены 8 – 16*

## **5. Блок заданий.**

### **Практические задания по материалу занятия.**

К началу демонстрации блока заданий ученики должны занять рабочие места и запустить Python (терминал IDLE) на своих компьютерах. Блок включает 3 практических задания с таймером, у которого время определяется сложностью задания; после завершения работы таймера демонстрируется разбор задания.

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

*\*см. сцены 17 – 22*

## **6. Рефлексия. Сообщение домашнего задания.**

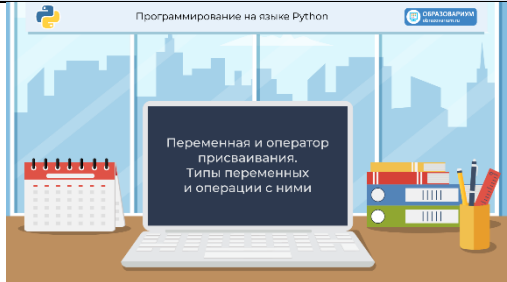
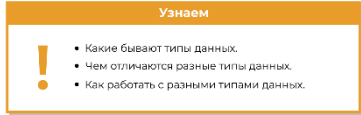
Завершаем демонстрацией ролика и кратким обобщением материалов занятия. Подводятся итоги и делаются выводы. Преподаватель дает ученикам домашнее задание к следующему занятию (*Приложение 2*).

*\*см. сцену 23*

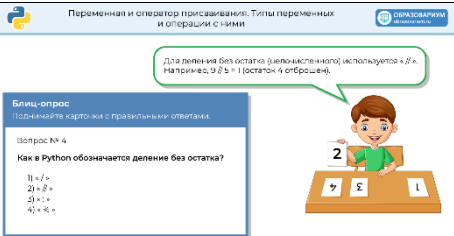
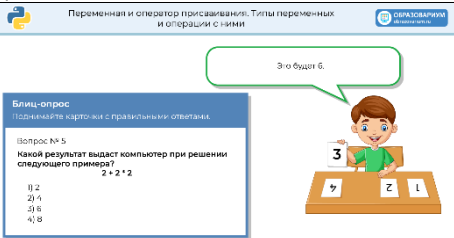
## Сценарный план видеоролика

В таблице «Содержание видеоролика» представлен дикторский 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.  Переменная и оператор присваивания. Типы переменных и операции с ними.		1 2
	При необходимости, останавливаем трансляцию и дополнительно комментируем.  На сегодняшнем уроке мы узнаем: какие бывают данные для компьютерных программ, чем они отличаются друг от друга и как можно с ними работать.	Сцена 1 	Сцена 2

<p>Блок повторения.</p> <p><b>Блиц-опрос</b></p>	<p><i>Повторение материала предыдущего урока; на столе имеются пронумерованные карточки; после каждого вопроса выбираем ту, номер которой, совпадает с правильным ответом. См. ответы на рисунке справа.</i></p> <p><b>Первый вопрос. Что такое АЛГОРИТМ?</b></p> <ol style="list-style-type: none"> <li>1) язык программирования</li> <li>2) последовательность действий</li> <li>3) служебная команда</li> <li>4) должность программиста</li> </ol> <p><i>Ответ № 2: алгоритм – последовательность действий для решения задачи</i></p>	<p>Переменная и оператор присваивания; Типы переменных и операции с ними</p> <p>Алгоритм – последовательность действий для решения задачи</p> <p><b>Блиц-опрос</b> Поднимайте карточки с правильными ответами.</p> <p>Вопрос № 1 Что такое АЛГОРИТМ?</p> <ol style="list-style-type: none"> <li>1) язык программирования</li> <li>2) последовательность действий</li> <li>3) служебная команда</li> <li>4) должность программиста</li> </ol> <p>Сцена 3</p>	<p>3</p> <p>4</p> <p>5</p> <p>6</p> <p>7</p>
	<p><b>Второй вопрос. Какой алгоритм не существует?</b></p> <ol style="list-style-type: none"> <li>1) линейный</li> <li>2) разветвленный</li> <li>3) конструктивный</li> <li>4) циклический</li> </ol> <p><i>Ответ № 3: не существует конструктивного алгоритма</i></p>	<p>Переменная и оператор присваивания; Типы переменных и операции с ними</p> <p>Не существует конструктивного алгоритма.</p> <p><b>Блиц-опрос</b> Поднимайте карточки с правильными ответами.</p> <p>Вопрос № 2 Какой алгоритм не существует?</p> <ol style="list-style-type: none"> <li>1) линейный</li> <li>2) разветвленный</li> <li>3) конструктивный</li> <li>4) циклический</li> </ol> <p>Сцена 4</p>	
	<p><b>Третий вопрос. Как в Python обозначается возведение в степень?</b></p> <ol style="list-style-type: none"> <li>1) «+»</li> <li>2) «*»</li> <li>3) «/»</li> <li>4) «**»</li> </ol> <p><i>Ответ № 4: взведение в степень это «**». Например, 4 ** 2</i></p>	<p>Переменная и оператор присваивания; Типы переменных и операции с ними</p> <p>Возведение в степень – это «**». Например, 4 ** 2.</p> <p><b>Блиц-опрос</b> Поднимайте карточки с правильными ответами.</p> <p>Вопрос № 3 Как в Python обозначается возведение в степень?</p> <ol style="list-style-type: none"> <li>1) «+»</li> <li>2) «*»</li> <li>3) «/»</li> <li>4) «**»</li> </ol> <p>Сцена 5</p>	

	<p><b>Четвертый вопрос.</b> Как в Python обозначается деление без остатка?</p> <ol style="list-style-type: none"> <li>1) «/»</li> <li>2) «//»</li> <li>3) «:»</li> <li>4) «%»</li> </ol> <p><i>Ответ № 2: для деления без остатка (целочисленного) используется «//»</i>  <i>Например, <math>9 // 5 = 1</math> (остаток 4 пропадет)</i></p>	<div data-bbox="1509 228 1960 464">  </div> <p>Сцена 6</p>
	<p><b>Пятый вопрос.</b> Какой результат выдаст компьютер при решении следующего примера?      <math>2 + 2 * 2</math></p> <ol style="list-style-type: none"> <li>1) 2</li> <li>2) 4</li> <li>3) 6</li> <li>4) 8</li> </ol> <p><i>Ответ № 3: это будет 6</i></p>	<div data-bbox="1509 560 1960 798">  </div> <p>Сцена 7</p>



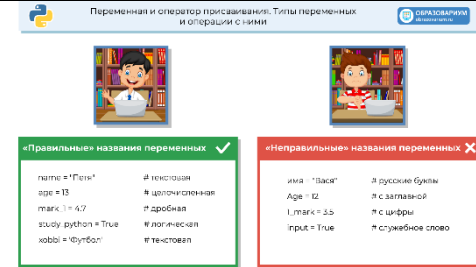


<p><b>Правила</b> при создании переменных:</p> <ul style="list-style-type: none"><li>Имя переменной пишется английскими буквами</li><li>Первая буква всегда прописная.</li><li>Переменная может содержать цифры.</li><li>Между словами имени переменной ставится нижнее подчеркивание</li></ul>	<div><div><div>Python</div><div>Переменная и оператор присваивания. Типы переменных и операции с ними</div><div>Слайдов: 10</div></div><div><div>Правила создания переменных</div><ul style="list-style-type: none"><li>Имя переменной пишется английскими буквами.</li><li>Первая буква всегда прописная. Пример: <code>name</code>, <code>age</code>, <code>number</code>.</li><li>Переменная может содержать цифры. Пример: <code>number1</code>, <code>text2</code>.</li><li>Между словами имени переменной ставится нижнее подчеркивание. Пример: <code>my_name</code>, <code>one_price</code>.</li></ul></div></div> <p>Сцена 10</p>																				
<p>Имя переменной должно обозначать ту информацию, которую она содержит.</p> <p><b>Например</b>, в программе данные об учениках класса.</p> <p>Переменную для хранения имени можно назвать <b>name</b>; переменную для возраста назовем <b>age</b>; переменную для среднего бала за первое полугодие – <b>mark_1</b>; изучает ли python запишем в переменную <b>study_python</b>.</p> <p>Можно использовать <b>русское слово</b>, написанное <b>английскими буквами</b>.</p>	<div><div><div>Python</div><div>Тип данных, работа с разными типами данных</div><div>Слайдов: 10</div></div><div></div></div> <p>Сцена 11</p>																				
<p><b>«Правильные»</b> названия переменных</p> <table><tr><td><code>name = "Петя"</code></td><td># текстовая</td></tr><tr><td><code>age = 13</code></td><td># целочисленная</td></tr><tr><td><code>mark_1 = 4.7</code></td><td># дробная</td></tr><tr><td><code>study_python = True</code></td><td># логическая</td></tr><tr><td><code>hobby = 'Футбол'</code></td><td># текстовая</td></tr></table>	<code>name = "Петя"</code>	# текстовая	<code>age = 13</code>	# целочисленная	<code>mark_1 = 4.7</code>	# дробная	<code>study_python = True</code>	# логическая	<code>hobby = 'Футбол'</code>	# текстовая	<div><div><div>Python</div><div>Переменная и оператор присваивания. Типы переменных и операции с ними</div><div>Слайдов: 10</div></div><div><div><div>«Правильные» названия переменных</div><div><table><tr><td><code>name = "Петя"</code></td><td># текстовая</td></tr><tr><td><code>age = 13</code></td><td># целочисленная</td></tr><tr><td><code>mark_1 = 4.7</code></td><td># дробная</td></tr><tr><td><code>study_python = True</code></td><td># логическая</td></tr><tr><td><code>hobby = 'Футбол'</code></td><td># текстовая</td></tr></table></div></div></div><div></div></div> <p>Сцена 12</p>	<code>name = "Петя"</code>	# текстовая	<code>age = 13</code>	# целочисленная	<code>mark_1 = 4.7</code>	# дробная	<code>study_python = True</code>	# логическая	<code>hobby = 'Футбол'</code>	# текстовая
<code>name = "Петя"</code>	# текстовая																				
<code>age = 13</code>	# целочисленная																				
<code>mark_1 = 4.7</code>	# дробная																				
<code>study_python = True</code>	# логическая																				
<code>hobby = 'Футбол'</code>	# текстовая																				
<code>name = "Петя"</code>	# текстовая																				
<code>age = 13</code>	# целочисленная																				
<code>mark_1 = 4.7</code>	# дробная																				
<code>study_python = True</code>	# логическая																				
<code>hobby = 'Футбол'</code>	# текстовая																				

Примеры «неправильных» переменных. Какие здесь совершены ошибки?

имя = "Вася" # русские буквы  
Age = 12 # с заглавной  
1\_mark = 3.5 # с цифры  
input = True # служебное слово

Переменная и оператор присваивания. Типы переменных и операции с ними



«Правильные» названия переменных ✓

- name = "Петя" # текстовая
- age = 13 # целочисленная
- mark\_1 = 4.7 # дробная
- study\_python = True # логическая
- hobby = "футбол" # текстовая

«Неправильные» названия переменных ✗

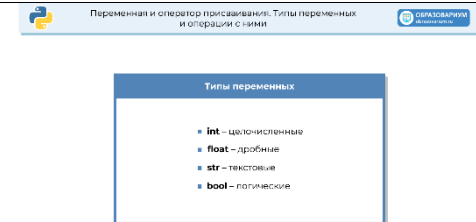
- имя = "Вася" # русские буквы
- Age = 12 # с заглавной
- 1\_mark = 3.5 # с цифры
- input = True # служебное слово

Сцена 13

Существует **четыре** типа переменных:

**int** - целочисленные  
**float** - дробные  
**str** - текстовые  
**bool** - логические.

Переменная и оператор присваивания. Типы переменных и операции с ними



Типы переменных

- **int** – целочисленные
- **float** – дробные
- **str** – текстовые
- **bool** – логические

Сцена 14

В Python, в отличие от многих других языков программирования, переменная может менять свой тип. От типа переменной зависит какие операции с ней можно производить и каким образом организуется ввод данных.

Для **целочисленной и дробной** – это просто число. Причем дробное число вводится с использованием ТОЧКИ, а не ЗАПЯТОЙ!

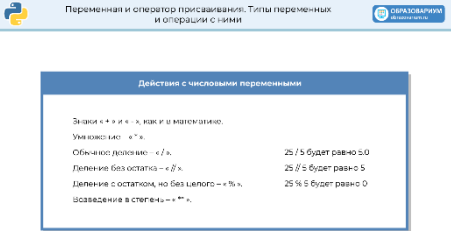
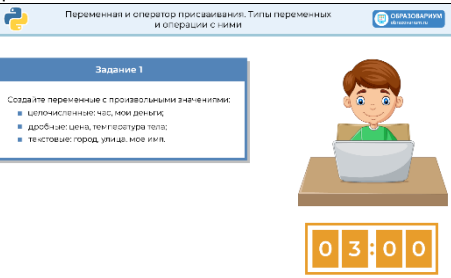
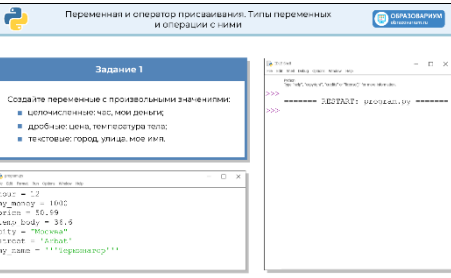
**Логические** переменные имеют значения True (правда) и False (ложь) и используются реже, внутри программы они равны числам один и ноль.

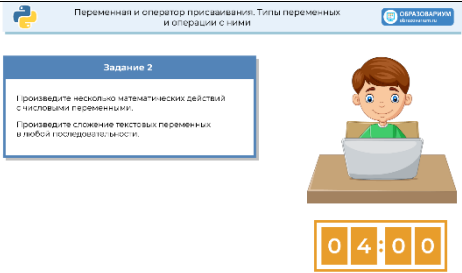
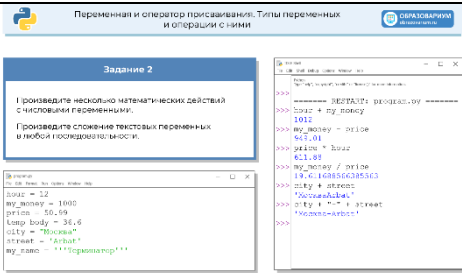
Для **текстовой** переменной всегда необходимы кавычки; в Python – одинарные или двойные (важно, что одинаковые); в текстовую переменную можно записывать и числа, просто они будут восприниматься как текст.

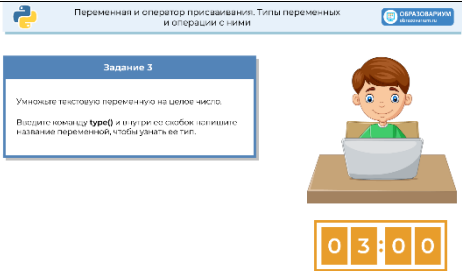
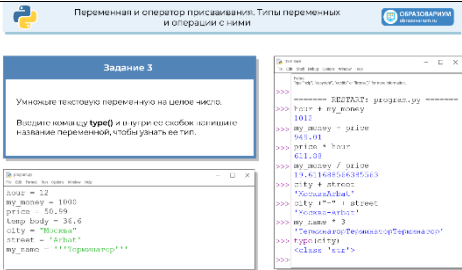
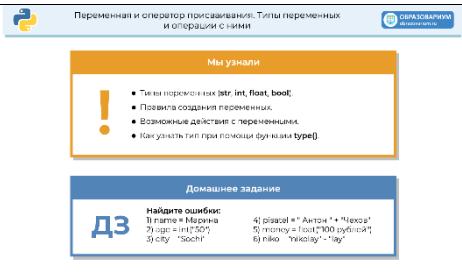
Переменная и оператор присваивания. Типы переменных и операции с ними

Тип	Пример	Что можно делать
Целочисленные	age = 13	Все математические действия.
Дробные	mark_1 = 4.7 mark_2 = 4.0	Все математические действия.
Логические	study_python = True play_chess = False	Все математические действия.
Текстовые	name = "Петя" sport = "футбол" my_class = "7А"	Складывать между собой. Складывать с другими символами. Умножать на целое число.

Сцена 15

	<p><b>Действия с числовыми переменными.</b></p> <p>Все они являются числами, а значит с ними можно производить любые <b>математические действия</b>: сложение, умножение, деление, вычитание, возведение в степень, извлечение квадратного корня и так далее.</p> <p>При использовании <b>обычного деления</b> ответ всегда будет в виде десятичной дроби, даже если нет никакого остатка.</p>	 <p>Сцена 16</p>	
<p>Блок заданий.</p> <p><b>Практические задания по материалу занятия:</b></p> <p>Задание 1</p> <p>Задание 2</p> <p>Задание 3</p>	<p><i>После окончания дикторского текста запускается таймер на 3 мин.</i></p> <p><b>Задание 1.</b></p> <p>Откройте новый проект и создайте переменные с произвольными значениями:</p> <p><b>Целочисленные:</b> час, мои деньги,  <b>Дробные:</b> цена, температура тела  <b>Текстовые:</b> город, улица, моё имя</p> <p><i>Запустите программу и убедитесь, что нет ошибок.</i></p>	 <p>Сцена 17</p>	<p>17</p> <p>18</p> <p>19</p> <p>20</p> <p>21</p> <p>22</p> <p>22</p>
	<p><b>Разбор задания 1.</b> Ваша программа к первому заданию может выглядеть так.</p> <pre> hour = 12 my_money = 1000 price = 50.99 temp_body = 36.6 city = "Москва" street = 'Arbat' my_name = ""Терминатор"" </pre> <p><b>Обратите внимание</b> на разные кавычки у текстовых переменных. Внутри кавычек информация может быть на любом языке.</p> <p>В <b>дробных</b> переменных используем точку, а не запятую.</p>	 <p>Сцена 18</p>	

	<p><i>После окончания дикторского текста запускается таймер на 4 мин.</i></p> <p><b>Задание 2.</b> Переменные теперь находятся в памяти компьютера. Перейдите в терминал и сделайте следующие действия:</p> <p>Произведите четыре математических действия с целочисленными и дробными переменными в любом сочетании (сложение, вычитание, умножение, деление)</p> <p>Произведите сложение текстовых переменных в любой последовательности.</p>	 <p>Сцена 19</p>
	<p><b>Разбор задания 2.</b> После второго задания на вашем терминале могут быть подобные варианты.</p> <div style="display: flex; justify-content: space-between;"> <div style="width: 30%;"> <p>hour + my_money 1012</p> <p>my_money - price 949.01</p> <p>price * hour 611.88</p> <p>my_money / price 19.611688566385563</p> <p>city + street 'МоскваArbat'</p> <p>city + "-" + street 'Москва-Arbat'</p> </div> <div style="width: 65%;"> <p>Как видим все математические действия с числовыми переменными работают.</p> <p>Результат зависит от введенной информации.</p> <p>Что происходит с текстовыми: их можно складывать друг с другом или с символами.</p> <p>В этом случае мы получим новое слово, состоящее из тех данных, которые были задействованы.</p> <p>Это как будто к бумажному листу, на котором написано слово поднесли лист с другим словом и теперь мы их читаем подряд.</p> </div> </div>	 <p>Сцена 20</p>

	<p><i>После окончания дикторского текста запускается таймер на 4 мин.</i></p> <p><b>Задание 3.</b> Умножьте текстовую переменную на целое число.</p> <p>Введите в терминале команду <b>type</b> и внутри ее скобок напишите название любой своей переменной, чтобы узнать ее тип.</p>	 <p>Сцена 21</p>	
	<p><b>Разбор задания 3.</b> После второго задания на вашем терминале могут быть подобные варианты.</p> <pre>my_name * 3 'ТерминаторТерминаторТерминатор' type (city) &lt;class 'str'&gt;</pre> <p>Текст можно умножать на целое число. В этом случае мы получим дублирование информации. Попробка отнять что-то от текстовой переменной или разделить ее на число приведет к сообщению об ошибке.</p> <p>Последняя строка – это результат работы команды <b>type</b>, которая выводит тип переменной <b>city</b>.</p>	 <p>Сцена 22</p>	
<p>Блок завершения занятия.</p> <p><b>Рефлексия.</b></p> <p><b>Сообщение домашнего задания</b></p>	<p><i>Завершаем демонстрацией ролика и кратким обобщением материалов занятия.</i></p> <p><b>Подведем итоги:</b></p> <ul style="list-style-type: none"> <li>- узнали о такой структуре для хранения информации как переменная;</li> <li>- познакомились с четырьмя типами переменных, с правилами их создания, с основными действиями при работе с переменными, как можно использовать переменные в терминале и узнавать их тип данных.</li> </ul> <p><i>Преподаватель дает ученикам домашнее задание к следующему занятию (Приложение 2).</i></p>	 <p>Сцена 23</p>	<p>23</p>

### Домашнее задание

Найдите ошибки:

- 1) name = Марина
- 2) age = int ("50")
- 3) city = "Sochi"
- 4) pisatel = " Антон " + "Чехов"
- 5) money = float ("100 рублей")
- 6) niko = "nikolay" - "lay"

Ответы: 1, 5, 6

### Практика

Проект «Дата»

В коде программы задайте исходные данные в переменных разных типов. Вы можете использовать следующие переменные или придумать свои.

```
text1 = "Последний день года - "  
day = 31  
month = " декабря "  
text2 = " не существует!"
```

Используя только эти переменные, придумайте и составьте два информационных сообщения. Сообщения запишите в переменные **info1**, **info2**. После завершения работы над кодом запустите программу и выведите в терминале значения переменных **info1** и **info2**.

## Краткие организационно-методические рекомендации по организации работы на занятии

«Переменная и оператор присваивания. Типы переменных и операции с ними».

**Перед началом** ролика педагог должен спросить, что ребята запомнили из темы «**алгоритмы**» и попросить привести собственные примеры алгоритмов из окружающей жизни. Желательно чтобы это были все три вида: линейный, с условием, циклический.

**Перед просмотром блока повторения** из ролика необходимо раздать дидактический материал для выполнения заданий (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.

**Во время теоретического блока** можно написать на флип-чарте ключевую информацию о названиях типов переменных, для чего ролик можно поставить на паузу.

**После теоретического блока** можно уточнить – весь ли материал понятен и, если возникнут вопросы – ответить на них до начала практической части.

**Перед блоком заданий № 1** необходимо проследить, чтобы у всех был открыт компьютерный терминал для практической работы. После каждого задания необходимо чтобы ребята запускали свой код и отслеживать возможные ошибки в написании. В случае их появления рекомендуется объяснить почему они появились.

**После просмотра ролика** – если остается время – можно сделать самостоятельно аналогичные проекты в терминале или проверить качество усвоенного теоретического материала.