|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Класс: 3** | |  |  | | | |
| **Тема урока** | | Сценарий игры | | | | |
| **Цели обучения, которые достигаются на данном уроке (ссылка на учебную программу)** | | 3.4.2.1 разрабатывать игру по готовому сценарию;  3.2.1.1 создавать маркированные, нумерованные списки | | | | |
| **Цель урока** | | помочь учащимся усвоить такие понятия как - переменные, типы переменных, типы данных, комментарии к программе, необходимые для осваивания основ программирования. | | | | |
| Ход урока | | | | | | |
| **Этапы урока** | **Деятельность учителя** | | | **Деятельность обучающихся** | **Оценивание** | **Ресурсы** |
| Организационный этап | Психологический настрой  Приветствие “Здравствуйте!”  Учащиеся поочередно касаются одноименных пальцев рук своего соседа, начиная с больших пальцев и говорят:  • желаю (соприкасаются большими пальцами);  • успеха (указательными);  • большого (средними);  • во всём (безымянными);  • и везде (мизинцами);  Здравствуйте! (прикосновение всей ладонью)  **Актуализация знаний.** На прошлом уроке мы научились объявлять переменные, и написали простую программу. Получили понятие о константах, идентификаторах и служебных словах. Для понимания того, как работает программа, необходимо познакомиться с понятием - переменные. | | | Показывают свои знание, при возникновении вопросов разбирают с учителем |  | Диалогическое обучение  Саморегулируемое обучение  Критическое мышление |
| Изучение нового материала | Любая программа обрабатывает некоторые данные. Данные могут быть представлены только как *константы* или *переменные*, причём имеющие собственные идентификаторы (имена). Как уже говорилось, рекомендуется давать имена отражающие смысл *константы* или *переменной.*                Переменные– одно из главных понятий в программировании. Для того, чтобы разбираться в программировании необходимо иметь чёткое представление о том, что такое переменная, как и где она хранится, и что с ней происходит в процессе выполнения программы.  *Переменной* называется величина, значение которой меняется в процессе исполнения алгоритма.  *Переменные* – это некоторые данные, обрабатываемые в программе и имеющие имя.  Как вы знаете, данные хранятся и обрабатываются в памяти компьютера. При работе программы – в оперативной памяти, а при выключении сохраняются в постоянной памяти. При создании программ используются разные типы данных, т.е. переменные различного типа. Это могут быть числа, символы, текст, логические переменные, процедуры, даты и др., которые, в свою очередь, могут подразделяться на определённые виды. Например, числовые данные могут быть целого типа, с дробной частью и т.д. В зависимости от типа данных, программа после объявления переменных, выделяет определённое количество ячеек в памяти, для хранения этих переменных. То есть, этим ячейкам присваиваются имена переменных и в этих ячейках, затем хранятся значения этих переменных. Храниться они могут или до конца выполнения программы, или до тех пор, пока переменной не присвоится другое значение. Имя переменной остается неизменным до конца программы, а значение самой переменной может меняться. В ЯП Паскаль обязательное объявление переменных, с описанием их имён, защищает программы от ошибок и повышает их надежность.  Раздел описания переменных начинается со служебного слова var, после которого следуют элементы описания. Переменные могут описываться как в начале программы, так и непосредственно внутри любого блока  begin … end. Внутриблочные описания переменных имеют тот же вид, что и в разделе описаний.  begin   var a1,a2,a3: integer; end.  Кроме того, переменные-параметры цикла могут описываться в заголовке оператора for.  Имена переменных в списке перечисляются через запятую. Например:  var   a,b: integer;   c,d: real;   m,n: byte;   s,s1: string;   ch,ch1: char;  f: boolean;  **Типы переменных.**  В зависимости от версии языка программирования Pascal типы переменных могут немного различаться. В программах написанных на ЯП **PascalABC** используются данные следующих типов:  integer (целый)  byte (байтовый) real (вещественный)  complex (комплексный)  string (строковый)  char (символьный) boolean (логический)  тип "массив" процедурный файловый классовый и некоторые другие.  Типы в **PascalABC** подразделяются на простые, строковые, структурированные, типы указателей, процедурные и классовые.  К ***простым*** относятся числовые (целые и вещественные) типы, логический, символьный, перечислимый и диапазонный тип.  Перечислимый тип данных задается перечислением всех значений, которые может принимать переменная данного типа. При описании отдельные значения указываются через запятую, а весь список заключается в круглые скобки. Например: Var Mesyac: (May,June,July,August );  ***Структурированные*** типы (т.е. имеющие какую-то структуру), образовываются массивами, записями, множествами и файлами.  Все простые типы, кроме вещественного, являются ***порядковыми***. Значения только этих типов могут быть *индексами* переменных и массивов и параметрами цикла for. *Индекс* – это порядковый номер в последовательности. Обычно обозначается символом i . Нумерация начинается с единицы. Например: В последовательности A,B,C…Z , индексы символов соответственно 1,2,3…26.  Если индекс iвыходит за пределы длины строки, то при выполнении программы появляется сообщение об ошибке.  **Типы данных:** 1. Порядковые целые.    |  |  |  |  | | --- | --- | --- | --- | | Имя типа | значение | Размер, байт | тип | | byte | 0..255 | 1 | числовой беззнаковый целый | | word | 0..65535 | 2 | числовой беззнаковый целый | | integer | -2147483648.. -2147483647 | 4 | числовой знаковый целый | | char | все символы в кодировке ASCII | 1 | символьный |   К порядковым относятся также *перечислимый* и *интервальный* тип.  *Перечислимый* тип определяется упорядоченным набором идентификаторов. Например:  type  Season = (Winter,Spring,Summer,Autumn);   DayOfWeek = (Mon,Tue,Wed,Thi,Thr,Sat,Sun);  Значения перечислимого типа занимают 4 байта.  *Интервальный* тип представляет собой подмножество значений целого, символьного или перечислимого типа и описывается в виде a..b,  где a - нижняя, b - верхняя граница интервального типа:  var    a: 0..10;   c: 'a'..'n';   d: Mon..Sat;  Тип, на основе которого строится интервальный тип, называется базовым для этого интервального типа. Значения интервального типа занимают 4 байта. 2.Вещественный тип. Тип real (числовой вещественный). Значения вещественного типа занимают 8 байт, содержат 15-16 значащих цифр и находятся в диапазоне -1.8∙10308 .. 1.8∙10308. Константы типаrealможно записывать как в форме с плавающей точкой, так и в экспоненциальной форме:   1.7;   0.013;  2.5e3 (2500);  1.4e-1 (0.14).  **3.Логический тип.**  Тип boolean (логический). Переменные и константы логического типа занимают 1 байт и могут иметь одно из двух значений, задаваемых константами True (истина - 1) иFalse (ложь - 0).  **4.Строковый тип.**  Тип string (строковый). Применяется при использовании текстовых данных в программе, состоит из набора последовательно расположенных символов char.По умолчанию под переменную типа string отводится 256 байт, при этом в нулевом байте хранится длина строки. Т.е. строки состоят, не более чем, из 255 символов. Пример описания:  var s: string;  Можно явно указать количество символов для переменной в [ ]. Например:  var s: string[50]; В данном случае под переменную выделяется 50 символов. Допускается при записи выражений строкового типа применять данные, символьного типа (char). В этом случае эти данные воспринимаются как string. К отдельным символам строкового типа можно обратиться по номеру этого символа в строке, аналогично индексу в массивах ( см. раздел 4 Массивы). **2.8. Комментарии к программе.**  В программе может присутствовать текст написанный разработчиком для пояснения к программе. Этот текст называется “комментарием к программе”. Даже опытные программисты считают необходимым присутствие комментариев в программах.  Комментарии заключаются между *скобками* { …}, (\*…\*) или пишутся после символов // (слеш). Комментарии не воспринимаются компьютером и не обрабатываются программой: { это - комментарий } (\* это - тоже комментарий \*)  // это - тоже комментарий  **Вопросы:** • Что такое переменная? • Какие типы переменных вы знаете? • Какие типы данных вы знаете? • В чём разница между простыми и структурированными типами? • Что называется индексом переменной?  **. Практическая часть.**  На прошлых уроках вы разобрались со структурной схемой программы на ЯП PascalABC. Сегодня на практической части мы научимся объявлять переменные, и напишем простую программу.  Program StObed;  var a,b,c,s:real;  begin  writeln('стоимость первого блюда');  writeln('стоимость второго блюда');  writeln('стоимость третьего блюда');  readln(a,b,c);  s:=a+b+c;  write('стоимость обеда ',s,' руб.');  readln;  end.  Исправьте программу так, чтобы в результате выполнения программы на экран выводилось ровно 4 строки вместе с ответом. | | |  | Словесная оценка учителя  . Взаимооценивание  **Стратегия«Стикер** | Критическое мышление.  Саморегулируемое обучение (самонаправленность в процессе работы над заданиями). |
| Рефлексия | *Ответить на вопросы*  Какие существуют виды операторов?  Что такое ошибка оператора?  Какие факторы влияют на оператора?  Перечислите требования безопасности перед работой за ПК?  Перечислите требования безопасности во время работы за ПК?  Перечислите требования безопасности после работы за ПК?  Перечислите требования безопасности во время аварийной ситуации ПК?  **Рефлексия**  • На уроке мне было важно и интересно…  • Сегодня на уроке я понял…  • Хотелось бы узнать… | | | Учащиеся подытоживают свои знания по изучаемой теме. |  |  |