

Variables and their declaration in Pascal

If you're new to Pascal, then take a look at the variables and declaration methods below to gain more useful knowledge, helping you understand and use Pascal easily and effectively.

Variables are values that can be changed within a program. Each variable in Pascal has a specific data type, which defines its size and memory layout, the range of values that can be stored in that memory, and the operators that can be applied to the variable.

Variable names can include letters, numbers, and underscores, and must begin with a letter or an underscore. Pascal is not case-sensitive, so using uppercase or lowercase makes no difference.



How to declare variables in Pascal

Pascal has many different functions; mastering them all is not something that can be done quickly, but requires considerable study. Many Pascal functions are simple, while others are complex, much like variables.

Article contents:

1. Basic variables in Pascal.
2. Declaring variables in Pascal.
3. Initializing variable values in Pascal.

Variables and their declaration in Pascal

1. Basic variables in Pascal

Các biến cơ bản trong Pascal	
Thực hiện bởi Taimienphi.vn	
Biến	Mô tả
Character	Thường là một octet (1 byte). Đây là kiểu dữ liệu số nguyên
Integer	Kiểu số nguyên
Real	Kiểu số thực
Boolean	Các giá trị logic True hoặc False. Đây cũng là kiểu dữ liệu số nguyên
Enumerated	Người dùng chỉ định danh sách
Subrange	Biểu diễn các biến có giá trị nằm trong 1 dải
String	Lưu trữ một mảng ký tự

The Pascal programming language also allows you to define different variable types. However, in this article, TipsMake will only introduce you to the basic variables.



2. Declaring variables in Pascal

All variables must be declared before using a Pascal program. All variable declarations follow the prefix `var`. A declaration specifies a list of variables, followed by a colon (:) and the variable type. The syntax for declaring variables is:

var

variable_list : type;

In this case, `type` must be a valid Pascal variable type, including characters, integers, real numbers, booleans, or any user-specified data type, etc. And `variable_list` can include one or more identifiers separated by commas.

Here are some valid variable declarations:

```
var  
age, weekdays : integer;  
taxrate, net_income: real;  
choice, isready: boolean;  
initials, grade: char;  
name, surname : string;
```

In previous articles, TipsMake mentioned that Pascal allows type declaration. The declared type can be defined by name or identifier. Additionally, type declarations can be used to define variable types.

For example:

```
type  
days, age = integer;  
yes, true = boolean;  
name, city = string;  
fees, expenses = real;
```

Type declarations can be used in variable declarations.

```
var  
weekdays, holidays : days;  
choice: yes;  
student_name, emp_name : name;  
capital: city;  
cost: expenses;
```

Note that there is a difference between type declaration and variable declaration. Type declarations specify types such as integer, real, etc. Variable declarations, on the other hand, specify the values that a variable can represent.

You can compare type declarations in Pascal with typedefs in C. The most important difference is that variable names refer to the memory location where the variable's value will be stored, whereas type declarations do not.

3. Initializing variable values in Pascal

Variables are assigned values using a colon (:) and an equals sign (=), followed by a constant expression. The general formula for assigning a value is:

```
variable_name := value;
```

By default, variables in Pascal are not initialized to zero and may contain garbage values. Therefore, it is better to initialize variables within the program.

Variables can be initialized (assigned initial values) in the variable declaration. Variable initialization follows the word `var` and the initialization syntax is as follows:

```
var  
variable_name : type = value;
```

Some examples include:

```
age: integer = 15;
taxrate: real = 0.5;
grade: char = 'A';
name: string = 'John Smith';
```

Below is an example of a complete program using variables and Pascal's declaration method:

```
program Greetings;
const
message = ' Welcome to the world of Pascal ';
type
name = string;
var
firstname, surname: name;
begin
writeln('Please enter your first name: ');
readln(firstname);
writeln('Please enter your surname: ');
readln(surname);
writeln;
writeln(message, ' ', firstname, ' ', surname);
end.
```

After being compiled and executed, the above code will produce the following result:

```
Please enter your first name:
John
Please enter your surname:
Smith
Welcome to the world of Pascal John Smith
```

Enumerated variables

You've seen how to use simple variable types like Integer, Real, and Boolean. Enumerated variables can be declared as follows:

```
var
var1, var2, . : enum-identifier;
```

When declaring enumeration variables, you can use type declarations. For example:

```
type
months = (January, February, March, April, May, June, July, August, September, October, November,
December);
Var
m: months;
.
M := January;
```

The following example illustrates the concept ?

```
program exEnumeration;  
type  
beverage = (coffee, tea, milk, water, coke, limejuice);  
var  
drink:beverage;  
begin  
writeln('Which drink do you want?');  
drink := limejuice;  
writeln('You can drink ', drink);  
end.
```

After being compiled and executed, the above code will produce the following result:

```
Which drink do you want?  
You can drink limejuice
```

Subdomain variables

Subdomain variables are declared as follows:

```
var  
subrange-name : lowerlim . upperlim;
```

Examples of declaring subdomain variables:

```
var  
marks: 1 . 100;  
grade: 'A' . 'E';  
age: 1 . 25;
```

The specific program uses variables of subdomain type:

```
program exSubrange;  
var  
marks: 1 . 100;  
grade: 'A' . 'E';  
begin  
writeln( 'Enter your marks(1 - 100): ');  
readln(marks);  
writeln( 'Enter your grade(A - E): ');  
readln(grade);  
writeln('Marks: ', marks, ' Grade: ', grade);  
end.
```

After being compiled and executed, the above code will produce the following result:

```
Enter your marks(1 - 100):  
100  
Enter your grade(A - E):
```

A

Marks: 100 Grade: A

In summary, a variable is a value that can change within a program. Each variable in Pascal has a specific data type. Above, TipsMake has introduced you to variables and how to declare them in Pascal, so you can grasp how to write functions in Pascal more easily and quickly become proficient. If you have any questions or need clarification, you can leave your comments in the section below the article.

You finished reading the article "**Variables and their declaration in Pascal**" edited by the [TipsMake](#) team. We hope this article has provided you with many useful tech tips and tricks. You can search for similar articles on tips and guides. Thank you for reading and for following us regularly.