

Welcome!

Welcome to **Pascal-Programming.info** - an educational website which is particularly developed to help you learn the Pascal Programming language step-by-step! Enjoy surfing

this website and get yourself started reading the lessons written purposely for those who would like to get themselves familiar and capable of writing structured Pascal programs.

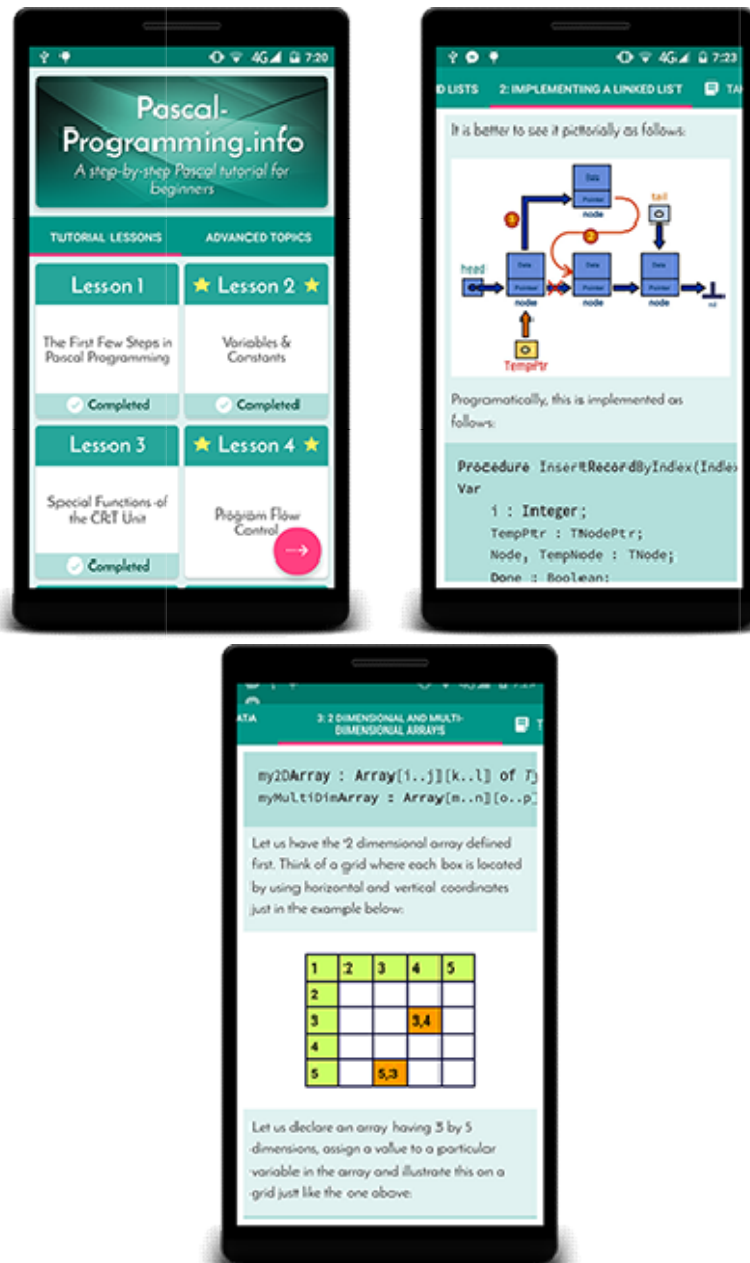
[Want to advertise here? Over 20,000 users per month.. Contact me!](#)

Download the Official Pascal-Programming.info App



The Official **Pascal-Programming.info** app is available for **FREE** on Google Play store! [Download now.](#)

Did you know that you can read all the updated lessons from the convenience of your phone/tablet? Yes that's right, you can read all the content **offline** from your device. The app is fully equipped with features to aid you through your learning process while tracking your progress from beginning to end. And the good news is that it is completely **free of charge!**



App Main Features:

- Read lessons and other further topics FREE & OFFLINE
- Take a quiz at the end of each tutorial lesson and monitor your learning progress
- Search for any content within the tutorial lessons and topics
- Take notes for anything you like to remember from each lesson



The Pascal Programming Language - Is it for me?

If you stepped into the site to start learning programming in Pascal, are new to programming, would like to start an adventure into the world of **computer programming**, then you are on the right track! There are no special pre-requisites to start learning Pascal.

Pascal is a relatively easy language, and helps novice programmers to introduce them to other languages such as C, C++, Java or C# - it is the doorstep to other high level programming languages (and stating from experience, I have first learnt Pascal which made it easy for me to migrate to other programming languages and I *highly recommend* fellow beginners to learn Pascal before other programming languages).

As for other dudes or dudettes who wish to study advanced material, we have another section for that! If you form part of this group then you can look at [Advanced Programming Concepts](#).

What is PASCAL?

Pascal is a high-level procedural programming language widely used as a as a language to learn general programming concepts. Sometimes Pascal is preferred to other languages, and could be useful to solve technical problems. It is not necessary to learn any other easier language to start learning Pascal, or any of that sort. It is a very easy programming language and helps you understand the basics of the world of programming. Also, it greatly helps you to start learning the C programming language and other language which are procedural. I had various experiences, one of which I learned Pascal, and then migrated to C programming very easily. The Pascal programming language has a programming structure and syntax similar to that of the C programming language. The successor language of Pascal is Delphi which is the object oriented version of Pascal.

Is the Pascal Programming Language still used?

Pascal is a structured procedural programming language which is practically the foundation of the more advanced object oriented programming language (called **Object Pascal**) of **Delphi**. Pascal contrasts with most modern day programming languages in that the latter uses 'Begin' and 'End' instead of '{', '}', 'And', 'Or', 'Not' instead of '&', '|', '!'; strict typing and many other differences. And this is why, perhaps, the language of Pascal is more welcomed by newbies because the syntax contains lots of English words and a program in pascal resembles much like an algorithm.

Delphi is used nowadays in developing various applications. You can write web applications, smartphone apps, desktop applications and lots more, using Delphi. Today, Delphi is owned by Embarcadero and is currently supported to keep up the pace with other modern programming languages in order to produce as much applications as other popular programming languages do.

A General Overview of Low-Level & High-Level Languages

There are two main categories of programming levels; these are low-level and high-level languages. The terms 'low' and 'high' does not mean that a group of 'low-level' languages are inferior to the 'high' ones. A low-level language consists of machine-oriented programming. For example, machine code is the only language with which the CPU understands. The other most-common low-level language, is the Assembly language. The Assembly language has a structure of machine-oriented programming. The Assembly Language, in other words, has a major control of the computer hardware. The Assembly Language makes use of mnemonics and has a better relationship with the memory and the CPU unit far more than that of other programming languages. The Assembly language is sometimes used as part of the source code written in Pascal. A high-level language consists of all those languages which are problem-oriented. A low-level language is not user-friendly, that is, makes it hard for the user to understand the structure of the program. Pascal, C/C++, BASIC, COBOL, Fortran, etc.. are examples of high-level programming languages. High-level languages represent their source code in such a way that it is more understandable to the programmer.



Machine code is a language which uses binary coding in its source codes. What do you understand by 0001001011 010101 1001110? It's machine code - the only language which is understandable by the CPU.

Interpreters and Compilers

Every high-level language has its own source which needs to be translated into the CPU's language - machine code. These translators are referred to as interpreters and compilers. A high-level language package contains either an interpreter, or a compiler, with which a high-level language uses to translate the source code into object code.

An **interpreter** is slower than a compiler, due to the fact that it has to be loaded into memory till the end of the translation of the program. It is less practical than the compiler but it is nonetheless used nowadays in critical applications such as browsers (JavaScript is an interpreted language). The interpreter can be helpful for finding and clearing errors in programs i.e. it is ideal for debugging.

A **compiler** is faster due to its single, complete translation of the program into object code. A compiler, compiles the source code (with no syntax errors) and creates an executable copy. This copy is called the 'object code'. After it has been compiled, it does not need a recompilation (unless the source code is changed and a recompilation is required to reflect the changes). The object code is used by the OS in order to execute the compiled program.

Where can I get a Pascal Source Code Editor & Compiler?

Back in the old days, Borland Turbo Pascal, was the one-stop shop Pascal compiler out there. However times change and so do our computers. This compiler has since been outdated and nowadays there are different compilers for Pascal one can use.

It is of fundamental importance that any learner who starts learning Pascal uses a compiler to compile their code and execute it while practicing Pascal coding. Fortunately, you can also find a number of online compilers which compile and execute the code on the go for you. At the time of writing, the best source code editor for Pascal programs and compiler is **Free Pascal**. It is a free compiler for running Pascal and Object Pascal programs. It is also a 32- and 64-bit Turbo Pascal and Delphi compatible Pascal compiler for Linux, Windows, OS/2, FreeBSD, Mac OS X, DOS and several other platforms.

When writing Pascal programs, you will be creating a pascal source file which in turn are saved as **.pas** files. These files are then compiled by your compiler and executed. Any text editor is suitable to be used to write Pascal programs.

One can download the Free Pascal IDE by [clicking here](#).

Ready to start?

Sit back, get yourself comfortable and when you are ready to start, hit **Lesson 1** from the lessons navigation menu on the top-left of the page or [click here](#) to get started.

Further Reading: Advanced Programming Concepts

Are you looking for some more advanced and applied concepts of the pascal programming language such as sorting, pointers and linked lists? Then the articles section is just the section for you. Articles provide discussions about advanced concepts of the language which you can integrate into your programs and use them wisely.



Why don't you have a look at the articles section by [clicking here](#) or choose an article from the table below:

Article #	Article Title
1.	Sorting
2.	The Stack Data Structure
3.	Pointers
4.	Linked Lists
5.	Recursion
6.	Object Oriented Programming - Part 1

Link to Pascal-Programming.info

It is highly appreciated for your help to get this site more known and let others benefit from the best pascal programming resource site on the net. Great effort was done to greatly improve this site's environment for every visitor. For those who would like to include a link to Pascal-Programming.info, they might use a nice logo like the one provided to you below linking to (right-click on the image and choose save as to save the image to your computer):



<http://www.pascal-programming.info/>
image

<http://pascal-programming.info/images/linktous.jpg>

<http://www.pascalprogramming.byethost15.com>

One may link to the
using:

Download a copy of Pascal-Programming.info Tutorials in PDF Format

Donate **€5** via PayPal and get an *offline PDF version* of this tutorial in return. Support Pascal-Programming.info with your generous donation!



[Click here to Donate & Get PDF](#)

New! Like Pascal-Programming.info on Facebook!

[Click here to visit Facebook Page.](#)