Perl 6

Perl 6 is a language specification for a new version of the programming language Perl. Any implementation that passes the official test suite can call itself "Perl 6". The most complete of the implementations at the moment is Rakudo Star (see http://rakudo.org/).

Other implementations at various levels of maturity and completeness (see http://perl6.org/compilers/):

- Niecza Perl 6 – a Perl 6 implementation focusing on optimization and efficient implementation research, mainly focused on Mono and .Net (see https://github.com/sorear/niecza).
- Perlito, also known as MiniPerl6 – a compiler collection that implements a (subset) of Perl 5 and Perl 6; it can compile Perl 5 and Perl 6 into another "backend language": Perl 5 to Perl 6 (v.v.), Perl 5/6 into Javascript, Perl 6 into Python 2.6, Ruby 1.9; Go, Common Lisp (see http://perlito.org/).

Perl 6 now runs on three virtual machines: Parrot, JVM and MoarVM. MoarVM is now so far developed that Parrot no longer is needed, even though Parrot is more stable than the other virtual machines. MoarVM offers a wider support for specified features than Perl 6 on JVM, which in turn is more advanced than Perl 6 on Parrot. MoarVM and JVM also offer great speed improvements over Parrot. MoarVM is solely developed for Perl 6 Rakudo and not for other programming languages.

Why the '6' in Perl 6
Perl 6 is versatile, intuitive, and flexible. It embraces several paradigms like procedural, object oriented, and functional programming, and offers powerful tools for parsing text. Perl 6 is the newest member of the family of languages known as Perl. It represents a major break in syntactic and semantic compatibility from Perl 5, thus the increase from 5 to 6. However, this does not mean that Perl 5 is going away anytime soon. In fact, quite the opposite. Both Perl 5 and Perl 6 have very active developer communities which mold the languages.

Differences and similarities between Perl 5 and Perl 6
Perl 5 developers try to extend the language in various ways while keeping backwards compatibility with past versions of Perl. Perl 6 developers extend the language by adding new syntactic and semantic features that enable more power and expressiveness without the restriction of backward compatibility with Perl 5 or earlier versions.

Some might ask, "Why call it Perl if it's a different language?" Perl is both the vagaries of syntax and a philosophy (there's more than one way to do it; easy things should be easy, and hard things possible); Perl is custom (comprehensive testing, idioms); Perl is architectural edifice (the Comprehensive Perl Archive Network); and Perl is community (perl5-porters, perl6-language). Both Perl 5 and Perl 6 share these attributes to varying degrees. As well, Perl is syncretic. Just as Perl borrows good ideas from other languages, Perl 5 and Perl 6 share features.

Download, use
For complete instructions for downloading and installing Rakudo, see http://www.rakudo.org/how-to-get-rakudo

Source code releases are available from http://github.com/

A binary release for windows is available from http://sourceforge.net/projects/parrotwin32/files/

MoarVM:
http://www.moarvm.org/
Book “Using Perl 6”
A group of Perl 6 Radudo developers were maintaining an online book to teach you how to program in Perl 6. It is free and it is a work-in-progress. The last version of “Using Perl 6” is from 2012.05.23 and, though still useful, it is getting out of date and in need of a serious update. It is written by well-known Perl 6 developers: Jonathan L. Duff, Moritz Lenz, Carl Maks, Patrick Michaud & Jonathan Worthington.  

All of the examples in this book will run with the Rakudo Perl 6 Compiler, but they are in no way specific to Rakudo – any sufficiently advanced Perl 6 implementation can run them. Good luck and – as the Perl 6 community often says – have fun!

Run Rakudo Perl 6
To run a Perl 6 program with Rakudo, include the installation directory in your system PATH variable and issue a command like:

$ perl6 hello.pl

(this must look familiar to any Perl developer).

If you invoke the Rakudo compiler without an explicit script to run, it enters a small interactive mode that allows the execution of Perl 6 statements from the command line.

Getting involved
If you are inspired now and want to contribute to the Perl 6 community, there are some resources available to you.


IRC: the channel #perl6 on irc.freenode.net discusses all things Perl 6.

Mailing lists: if you need programming help with Perl 6, send an email to perl6-users@perl.org
For issues regarding the Perl 6 language specification, contact perl6-language@perl.org
For issues regarding Perl 6 compilers, send email to perl6-compiler@perl.org

Modules for Rakudo Perl 6
A growing list of modules is published on http://modules.perl6.org/ and at the moment of writing this, the number of modules exceeded 220. Of course, this can not (yet) be compared with the immense number of Perl 5 modules on CPAN. Nevertheless, number of modules grows every month and makes Perl 6 more useful. The module management tool for Rakudo Perl, panda, makes your life as a module user easier.

Earlier in 2014 it has been made possible to upload Perl 6 modules to CPAN. Quite some work is put into the efforts to make Perl 6 modules downloadable / installable from CPAN.

At YAPC::EU 2012, one of the Perl 6 developers exclaimed: “There is DarkPan for Perl 6”. Which means he found two independent, unknown, groups of people who published some Perl 6 modules on Github. Which means Perl 6 is in production (without the Perl 6 developers knowing about it). It is in use.

In 2015, Perl 6.0.0 will be released. The whole world can start using it. You can use it now already, so be an early adapter and show everybody how awesome programming can be.

Perl 6 is here. It exists. It is ready to be used. Enjoy.