Programming languages & Styles supported by the minted package

Overleaf

```
Pygments version 2.1.3, (c) 2006-2015 by Georg Brandl.

Lexers:
~~~~~~~
* abap:
    ABAP (filenames *.abap, *.ABAP)
* abnf:
    ABNF (filenames *.abnf)
* ada, ada95, ada2005:
    Ada (filenames *.adb, *.ads, *.ada)
* adl:
    ADL (filenames *.adl, *.adls, *.adlf, *.adlx)
* agda:
    Agda (filenames *.agda)
* ahk, autohotkey:
    autohotkey (filenames *.ahk, *.ahkl)
* alloy:
    Alloy (filenames *.als)
* antlr-as, antlr-actionscript:
    ANTLR With ActionScript Target (filenames *.G, *.g)
* antlr-cpp:
    ANTLR With CPP Target (filenames *.G, *.g)
* antlr-csharp, antlr-c#:
    ANTLR With C# Target (filenames *.G, *.g)
* antlr-java:
    ANTLR With Java Target (filenames *.G, *.g)
* antlr-objc:
    ANTLR With ObjectiveC Target (filenames *.G, *.g)
* antlr-perl:
    ANTLR With Perl Target (filenames *.G, *.g)
* antlr-python:
    ANTLR With Python Target (filenames *.G, *.g)
* antlr-ruby, antlr-rb:
    ANTLR With Ruby Target (filenames *.G, *.g)
* antlr:
    ANTLR
* apacheconf, aconf, apache:
    ApacheConf (filenames .htaccess, apache.conf, apache2.conf)
```

```
* apl:
   APL (filenames *.apl)
* applescript:
   AppleScript (filenames *.applescript)
* arduino:
   Arduino (filenames *.ino)
* as, actionscript:
   ActionScript (filenames *.as)
* as3, actionscript3:
   ActionScript 3 (filenames *.as)
* aspectj:
   AspectJ (filenames *.aj)
* aspx-cs:
   aspx-cs (filenames *.aspx, *.asax, *.ascx, *.ashx, *.asmx, *.axd)
* aspx-vb:
   aspx-vb (filenames *.aspx, *.asax, *.ascx, *.ashx, *.asmx, *.axd)
* asy, asymptote:
   Asymptote (filenames *.asy)
* at, ambienttalk, ambienttalk/2:
   AmbientTalk (filenames *.at)
* autoit:
   AutoIt (filenames *.au3)
* awk, gawk, mawk, nawk:
   Awk (filenames *.awk)
* basemake:
   Base Makefile
* bash, sh, ksh, shell:
   Bash (filenames *.sh, *.ksh, *.bash, *.ebuild, *.eclass, *.exheres-0, *.exlib,
       .bashrc, bashrc, .bash_*, bash_*, PKGBUILD)
* bat, batch, dosbatch, winbatch:
   Batchfile (filenames *.bat, *.cmd)
* bbcode:
   BBCode
* bc:
   BC (filenames *.bc)
* befunge:
   Befunge (filenames *.befunge)
* blitzbasic, b3d, bplus:
   BlitzBasic (filenames *.bb, *.decls)
* blitzmax, bmax:
   BlitzMax (filenames *.bmx)
* bnf:
   BNF (filenames *.bnf)
* boo:
   Boo (filenames *.boo)
* boogie:
   Boogie (filenames *.bpl)
* brainfuck, bf:
   Brainfuck (filenames *.bf, *.b)
* bro:
   Bro (filenames *.bro)
```

```
* bugs, winbugs, openbugs:
    BUGS (filenames *.bug)
* c-objdump:
    c-objdump (filenames *.c-objdump)
* c:
    C (filenames *.c, *.h, *.idc)
* ca65:
    ca65 assembler (filenames *.s)
* cadl:
    cADL (filenames *.cadl)
* camkes, idl4:
    CAmkES (filenames *.camkes, *.idl4)
* cbmbas:
    CBM BASIC V2 (filenames *.bas)
* ceylon:
    Ceylon (filenames *.ceylon)
* cfc:
    Coldfusion CFC (filenames *.cfc)
* cfengine3, cf3:
    CFEngine3 (filenames *.cf)
* cfm:
    Coldfusion HTML (filenames *.cfm, *.cfml)
* cfs:
    cfstatement
* chai, chaiscript:
    ChaiScript (filenames *.chai)
* chapel, chpl:
    Chapel (filenames *.chpl)
* cheetah, spitfire:
    Cheetah (filenames *.tmpl, *.spt)
* cirru:
    Cirru (filenames *.cirru)
* clay:
    Clay (filenames *.clay)
* clojure, clj:
    Clojure (filenames *.clj)
* clojurescript, cljs:
    ClojureScript (filenames *.cljs)
* cmake:
    CMake (filenames *.cmake, CMakeLists.txt)
* cobol:
    COBOL (filenames *.cob, *.COB, *.cpy, *.CPY)
* cobolfree:
    COBOLFree (filenames *.cbl, *.CBL)
* coffee-script, coffeescript, coffee:
    CoffeeScript (filenames *.coffee)
* common-lisp, cl, lisp:
    Common Lisp (filenames *.cl, *.lisp)
* componentpascal, cp:
    Component Pascal (filenames *.cp, *.cps)
* console, shell-session:
    Bash Session (filenames *.sh-session, *.shell-session)
* control, debcontrol:
    Debian Control file (filenames control)
* coq:
    Coq (filenames *.v)
* cpp, c++:
    C++ (filenames *.cpp, *.hpp, *.c++, *.h++, *.cc, *.hh, *.cxx, *.hxx, *.C, *.H,
        *.cp, *.CPP)
* cpp-objdump, c++-objdumb, cxx-objdump:
    cpp-objdump (filenames *.cpp-objdump, *.c++-objdump, *.cxx-objdump)
* cpsa:
    CPSA (filenames *.cpsa)
* crmsh, pcmk:
    Crmsh (filenames *.crmsh, *.pcmk)
* croc:
    Croc (filenames *.croc)
* cryptol, cry:
    Cryptol (filenames *.cry)
* csharp, c#:
    C# (filenames *.cs)
* csound, csound-orc:
    Csound Orchestra (filenames *.orc)
* csound-document, csound-csd:
    Csound Document (filenames *.csd)
* csound-score, csound-sco:
    Csound Score (filenames *.sco)
* css+django, css+jinja:
    CSS+Django/Jinja
* css+erb, css+ruby:
    CSS+Ruby
* css+genshitext, css+genshi:
    CSS+Genshi Text
* css+lasso:
    CSS+Lasso
* css+mako:
    CSS+Mako
* css+mozpreproc:
    CSS+mozpreproc (filenames *.css.in)
* css+myghty:
    CSS+Myghty
* css+php:
    CSS+PHP
* css+smarty:
    CSS+Smarty
* css:
    CSS (filenames *.css)
* cucumber, gherkin:
    Gherkin (filenames *.feature)
* cuda, cu:
    CUDA (filenames *.cu, *.cuh)
* cypher:
   Cypher (filenames *.cyp, *.cypher)
* cython, pyx, pyrex:
   Cython (filenames *.pyx, *.pxd, *.pxi)
* d-objdump:
   d-objdump (filenames *.d-objdump)
* d:
   D (filenames *.d, *.di)
* dart:
   Dart (filenames *.dart)
* delphi, pas, pascal, objectpascal:
   Delphi (filenames *.pas)
* dg:
   dg (filenames *.dg)
* diff, udiff:
   Diff (filenames *.diff, *.patch)
* django, jinja:
   Django/Jinja
* docker, dockerfile:
   Docker (filenames Dockerfile, *.docker)
* doscon:
   MSDOS Session
* dpatch:
   Darcs Patch (filenames *.dpatch, *.darcspatch)
* dtd:
   DTD (filenames *.dtd)
* duel, jbst, jsonml+bst:
   Duel (filenames *.duel, *.jbst)
* dylan-console, dylan-repl:
   Dylan session (filenames *.dylan-console)
* dylan-lid, lid:
   DylanLID (filenames *.lid, *.hdp)
* dylan:
   Dylan (filenames *.dylan, *.dyl, *.intr)
* earl-grey, earlgrey, eg:
   Earl Grey (filenames *.eg)
* easytrieve:
   Easytrieve (filenames *.ezt, *.mac)
* ebnf:
   EBNF (filenames *.ebnf)
* ec:
   eC (filenames *.ec, *.eh)
* ecl:
   ECL (filenames *.ecl)
* eiffel:
   Eiffel (filenames *.e)
* elixir, ex, exs:
   Elixir (filenames *.ex, *.exs)
* elm:
   Elm (filenames *.elm)
* emacs, elisp, emacs-lisp:
   EmacsLisp (filenames *.el)
```

```
* erb:
   ERB
* erl:
   Erlang erl session (filenames *.erl-sh)
* erlang:
   Erlang (filenames *.erl, *.hrl, *.es, *.escript)
* evoque:
   Evoque (filenames *.evoque)
* ezhil:
   Ezhil (filenames *.n)
* factor:
   Factor (filenames *.factor)
* fan:
   Fantom (filenames *.fan)
* fancy, fy:
   Fancy (filenames *.fy, *.fancypack)
* felix, flx:
   Felix (filenames *.flx, *.flxh)
* fish, fishshell:
   Fish (filenames *.fish, *.load)
* fortran:
   Fortran (filenames *.f03, *.f90, *.F03, *.F90)
* fortranfixed:
   FortranFixed (filenames *.f, *.F)
* foxpro, vfp, clipper, xbase:
   FoxPro (filenames *.PRG, *.prg)
* fsharp:
   FSharp (filenames *.fs, *.fsi)
* gap:
   GAP (filenames *.g, *.gd, *.gi, *.gap)
* gas, asm:
   GAS (filenames *.s, *.S)
* genshi, kid, xml+genshi, xml+kid:
   Genshi (filenames *.kid)
* genshitext:
   Genshi Text
* glsl:
   GLSL (filenames *.vert, *.frag, *.geo)
* gnuplot:
   Gnuplot (filenames *.plot, *.plt)
* go:
   Go (filenames *.go)
* golo:
   Golo (filenames *.golo)
* gooddata-cl:
   GoodData-CL (filenames *.gdc)
* gosu:
   Gosu (filenames *.gs, *.gsx, *.gsp, *.vark)
* groff, nroff, man:
   Groff (filenames *.[1234567], *.man)
* groovy:
  Groovy (filenames *.groovy, *.gradle)
* gst:
  Gosu Template (filenames *.gst)
* haml:
  Haml (filenames *.haml)
* handlebars:
  Handlebars
* haskell, hs:
  Haskell (filenames *.hs)
* haxeml, hxml:
  Hxml (filenames *.hxml)
* hexdump:
  Hexdump
* html+cheetah, html+spitfire, htmlcheetah:
  HTML+Cheetah
* html+django, html+jinja, htmldjango:
  HTML+Django/Jinja
* html+evoque:
  HTML+Evoque (filenames *.html)
* html+genshi, html+kid:
  HTML+Genshi
* html+handlebars:
  HTML+Handlebars (filenames *.handlebars, *.hbs)
* html+lasso:
  HTML+Lasso
* html+mako:
  HTML+Mako
* html+myghty:
  HTML+Myghty
* html+php:
  HTML+PHP (filenames *.phtml)
* html+smarty:
  HTML+Smarty
* html+twig:
  HTML+Twig (filenames *.twig)
* html+velocity:
  HTML+Velocity
* html:
  HTML (filenames *.html, *.htm, *.xhtml, *.xslt)
* http:
  HTTP
* hx, haxe, hxsl:
  Haxe (filenames *.hx, *.hxsl)
* hybris, hy:
  Hybris (filenames *.hy, *.hyb)
* hylang:
  Hy (filenames *.hy)
* i6t:
  Inform 6 template (filenames *.i6t)
* idl:
  IDL (filenames *.pro)
```

```
* idris, idr:
   Idris (filenames *.idr)
* iex:
   Elixir iex session
* igor, igorpro:
   Igor (filenames *.ipf)
* inform6, i6:
   Inform 6 (filenames *.inf)
* inform7, i7:
   Inform 7 (filenames *.ni, *.i7x)
* ini, cfg, dosini:
   INI (filenames *.ini, *.cfg, *.inf)
* io:
   Io (filenames *.io)
* ioke, ik:
   Ioke (filenames *.ik)
* irc:
   IRC logs (filenames *.weechatlog)
* isabelle:
   Isabelle (filenames *.thy)
* j:
   J (filenames *.ijs)
* jade:
   Jade (filenames *.jade)
* jags:
   JAGS (filenames *.jag, *.bug)
* jasmin, jasminxt:
   Jasmin (filenames *.j)
* java:
   Java (filenames *.java)
* javascript+mozpreproc:
   Javascript+mozpreproc (filenames *.js.in)
* jcl:
   JCL (filenames *.jcl)
* jlcon:
   Julia console
* js+cheetah, javascript+cheetah, js+spitfire, javascript+spitfire:
   JavaScript+Cheetah
* js+django, javascript+django, js+jinja, javascript+jinja:
   JavaScript+Django/Jinja
* js+erb, javascript+erb, js+ruby, javascript+ruby:
   JavaScript+Ruby
* js+genshitext, js+genshi, javascript+genshitext, javascript+genshi:
   JavaScript+Genshi Text
* js+lasso, javascript+lasso:
   JavaScript+Lasso
* js+mako, javascript+mako:
   JavaScript+Mako
* js+myghty, javascript+myghty:
   JavaScript+Myghty
* js+php, javascript+php:
    JavaScript+PHP
* js+smarty, javascript+smarty:
    JavaScript+Smarty
* js, javascript:
    JavaScript (filenames *.js, *.jsm)
* json:
    JSON (filenames *.json)
* jsonld, json-ld:
    JSON-LD (filenames *.jsonld)
* jsp:
    Java Server Page (filenames *.jsp)
* julia, jl:
    Julia (filenames *.jl)
* kal:
    Kal (filenames *.kal)
* kconfig, menuconfig, linux-config, kernel-config:
    Kconfig (filenames Kconfig, *Config.in*, external.in*, standard-modules.in)
* koka:
    Koka (filenames *.kk, *.kki)
* kotlin:
    Kotlin (filenames *.kt)
* lagda, literate-agda:
    Literate Agda (filenames *.lagda)
* lasso, lassoscript:
    Lasso (filenames *.lasso, *.lasso[89])
* lcry, literate-cryptol, lcryptol:
    Literate Cryptol (filenames *.lcry)
* lean:
    Lean (filenames *.lean)
* less:
    LessCss (filenames *.less)
* lhs, literate-haskell, lhaskell:
    Literate Haskell (filenames *.lhs)
* lidr, literate-idris, lidris:
    Literate Idris (filenames *.lidr)
* lighty, lighttpd:
    Lighttpd configuration file
* limbo:
    Limbo (filenames *.b)
* liquid:
    liquid (filenames *.liquid)
* live-script, livescript:
    LiveScript (filenames *.ls)
* llvm:
    LLVM (filenames *.ll)
* logos:
    Logos (filenames *.x, *.xi, *.xm, *.xmi)
* logtalk:
    Logtalk (filenames *.lgt, *.logtalk)
* lsl:
    LSL (filenames *.lsl)
```

```
* lua:
   Lua (filenames *.lua, *.wlua)
* make, makefile, mf, bsdmake:
   Makefile (filenames *.mak, *.mk, Makefile, makefile, Makefile.*, GNUmakefile)
* mako:
   Mako (filenames *.mao)
* maql:
   MAQL (filenames *.maql)
* mask:
   Mask (filenames *.mask)
* mason:
   Mason (filenames *.m, *.mhtml, *.mc, *.mi, autohandler, dhandler)
* mathematica, mma, nb:
   Mathematica (filenames *.nb, *.cdf, *.nbp, *.ma)
* matlab:
   Matlab (filenames *.m)
* matlabsession:
   Matlab session
* minid:
   MiniD
* modelica:
   Modelica (filenames *.mo)
* modula2, m2:
   Modula-2 (filenames *.def, *.mod)
* monkey:
   Monkey (filenames *.monkey)
* moocode, moo:
   MOOCode (filenames *.moo)
* moon, moonscript:
   MoonScript (filenames *.moon)
* mozhashpreproc:
   mozhashpreproc
* mozpercentpreproc:
   mozpercentpreproc
* mql, mq4, mq5, mql4, mql5:
   MQL (filenames *.mq4, *.mq5, *.mqh)
* mscgen, msc:
   Mscgen (filenames *.msc)
* mupad:
   MuPAD (filenames *.mu)
* mxml:
   MXML (filenames *.mxml)
* myghty:
   Myghty (filenames *.myt, autodelegate)
* mysql:
   MySQL
* nasm:
   NASM (filenames *.asm, *.ASM)
* nemerle:
   Nemerle (filenames *.n)
* nesc:
  nesC (filenames *.nc)
* newlisp:
  NewLisp (filenames *.lsp, *.nl)
* newspeak:
  Newspeak (filenames *.ns2)
* nginx:
  Nginx configuration file
* nimrod, nim:
  Nimrod (filenames *.nim, *.nimrod)
* nit:
  Nit (filenames *.nit)
* nixos, nix:
  Nix (filenames *.nix)
* nsis, nsi, nsh:
  NSIS (filenames *.nsi, *.nsh)
* numpy:
  NumPy
* objdump-nasm:
  objdump-nasm (filenames *.objdump-intel)
* objdump:
  objdump (filenames *.objdump)
* objective-c++, objectivec++, obj-c++, objc++:
  Objective-C++ (filenames *.mm, *.hh)
* objective-c, objectivec, obj-c, objc:
  Objective-C (filenames *.m, *.h)
* objective-j, objectivej, obj-j, objj:
  Objective-J (filenames *.j)
* ocaml:
  OCaml (filenames *.ml, *.mli, *.mll, *.mly)
* octave:
  Octave (filenames *.m)
* odin:
  ODIN (filenames *.odin)
* ooc:
  Ooc (filenames *.ooc)
* opa:
  Opa (filenames *.opa)
* openedge, abl, progress:
  OpenEdge ABL (filenames *.p, *.cls)
* pacmanconf:
  PacmanConf (filenames pacman.conf)
* pan:
  Pan (filenames *.pan)
* parasail:
  ParaSail (filenames *.psi, *.psl)
* pawn:
  Pawn (filenames *.p, *.pwn, *.inc)
* perl, pl:
  Perl (filenames *.pl, *.pm, *.t)
* perl6, pl6:
  Perl6 (filenames *.pl, *.pm, *.nqp, *.p6, *.6pl, *.p6l, *.pl6, *.6pm, *.p6m,
       *.pm6, *.t)
* php, php3, php4, php5:
    PHP (filenames *.php, *.php[345], *.inc)
* pig:
    Pig (filenames *.pig)
* pike:
    Pike (filenames *.pike, *.pmod)
* pkgconfig:
    PkgConfig (filenames *.pc)
* plpgsql:
    PL/pgSQL
* postgresql, postgres:
    PostgreSQL SQL dialect
* postscript, postscr:
    PostScript (filenames *.ps, *.eps)
* pot, po:
    Gettext Catalog (filenames *.pot, *.po)
* pov:
    POVRay (filenames *.pov, *.inc)
* powershell, posh, ps1, psm1:
    PowerShell (filenames *.ps1, *.psm1)
* praat:
    Praat (filenames *.praat, *.proc, *.psc)
* prolog:
    Prolog (filenames *.ecl, *.prolog, *.pro, *.pl)
* properties, jproperties:
    Properties (filenames *.properties)
* protobuf, proto:
    Protocol Buffer (filenames *.proto)
* ps1con:
    PowerShell Session
* psql, postgresql-console, postgres-console:
    PostgreSQL console (psql)
* puppet:
    Puppet (filenames *.pp)
* py3tb:
    Python 3.0 Traceback (filenames *.py3tb)
* pycon:
    Python console session
* pypylog, pypy:
    PyPy Log (filenames *.pypylog)
* pytb:
    Python Traceback (filenames *.pytb)
* python, py, sage:
    Python (filenames *.py, *.pyw, *.sc, SConstruct, SConscript, *.tac, *.sage)
* python3, py3:
    Python 3
* qbasic, basic:
    QBasic (filenames *.BAS, *.bas)
* qml, qbs:
    QML (filenames *.qml, *.qbs)
```

```
* qvto, qvt:
   QVTO (filenames *.qvto)
* racket, rkt:
   Racket (filenames *.rkt, *.rktd, *.rktl)
* ragel-c:
   Ragel in C Host (filenames *.rl)
* ragel-cpp:
   Ragel in CPP Host (filenames *.rl)
* ragel-d:
   Ragel in D Host (filenames *.rl)
* ragel-em:
   Embedded Ragel (filenames *.rl)
* ragel-java:
   Ragel in Java Host (filenames *.rl)
* ragel-objc:
   Ragel in Objective C Host (filenames *.rl)
* ragel-ruby, ragel-rb:
   Ragel in Ruby Host (filenames *.rl)
* ragel:
   Ragel
* raw:
   Raw token data
* rb, ruby, duby:
   Ruby (filenames *.rb, *.rbw, Rakefile, *.rake, *.gemspec, *.rbx, *.duby,
       Gemfile)
* rbcon, irb:
   Ruby irb session
* rconsole, rout:
   RConsole (filenames *.Rout)
* rd:
   Rd (filenames *.Rd)
* rebol:
   REBOL (filenames *.r, *.r3, *.reb)
* red, red/system:
   Red (filenames *.red, *.reds)
* redcode:
   Redcode (filenames *.cw)
* registry:
   reg (filenames *.reg)
* resource, resourcebundle:
   ResourceBundle (filenames *.txt)
* rexx, arexx:
   Rexx (filenames *.rexx, *.rex, *.rx, *.arexx)
* rhtml, html+erb, html+ruby:
   RHTML (filenames *.rhtml)
* roboconf-graph:
   Roboconf Graph (filenames *.graph)
* roboconf-instances:
   Roboconf Instances (filenames *.instances)
* robotframework:
   RobotFramework (filenames *.txt, *.robot)
```

```
* rql:
   RQL (filenames *.rql)
* rsl:
   RSL (filenames *.rsl)
* rst, rest, restructuredtext:
   reStructuredText (filenames *.rst, *.rest)
* rts, trafficscript:
   TrafficScript (filenames *.rts)
* rust:
   Rust (filenames *.rs, *.rs.in)
* sass:
   Sass (filenames *.sass)
* sc, supercollider:
   SuperCollider (filenames *.sc, *.scd)
* scala:
   Scala (filenames *.scala)
* scaml:
   Scaml (filenames *.scaml)
* scheme, scm:
   Scheme (filenames *.scm, *.ss)
* scilab:
   Scilab (filenames *.sci, *.sce, *.tst)
* scss:
   SCSS (filenames *.scss)
* shen:
   Shen (filenames *.shen)
* slim:
   Slim (filenames *.slim)
* smali:
   Smali (filenames *.smali)
* smalltalk, squeak, st:
   Smalltalk (filenames *.st)
* smarty:
   Smarty (filenames *.tpl)
* sml:
   Standard ML (filenames *.sml, *.sig, *.fun)
* snobol:
   Snobol (filenames *.snobol)
* sourceslist, sources.list, debsources:
   Debian Sourcelist (filenames sources.list)
* sp:
   SourcePawn (filenames *.sp)
* sparql:
   SPARQL (filenames *.rq, *.sparql)
* spec:
   RPMSpec (filenames *.spec)
* splus, s, r:
   S (filenames *.S, *.R, .Rhistory, .Rprofile, .Renviron)
* sql:
   SQL (filenames *.sql)
* sqlite3:
    sqlite3con (filenames *.sqlite3-console)
* squidconf, squid.conf, squid:
    SquidConf (filenames squid.conf)
* ssp:
    Scalate Server Page (filenames *.ssp)
* stan:
    Stan (filenames *.stan)
* swift:
    Swift (filenames *.swift)
* swig:
    SWIG (filenames *.swg, *.i)
* systemverilog, sv:
    systemverilog (filenames *.sv, *.svh)
* tads3:
    TADS 3 (filenames *.t)
* tap:
    TAP (filenames *.tap)
* tcl:
    Tcl (filenames *.tcl, *.rvt)
* tcsh, csh:
    Tcsh (filenames *.tcsh, *.csh)
* tcshcon:
    Tcsh Session
* tea:
    Tea (filenames *.tea)
* termcap:
    Termcap (filenames termcap, termcap.src)
* terminfo:
    Terminfo (filenames terminfo, terminfo.src)
* terraform, tf:
    Terraform (filenames *.tf)
* tex, latex:
    TeX (filenames *.tex, *.aux, *.toc)
* text:
    Text only (filenames *.txt)
* thrift:
    Thrift (filenames *.thrift)
* todotxt:
    Todotxt (filenames todo.txt, *.todotxt)
* trac-wiki, moin:
    MoinMoin/Trac Wiki markup
* treetop:
    Treetop (filenames *.treetop, *.tt)
* ts, typescript:
    TypeScript (filenames *.ts)
* turtle:
    Turtle (filenames *.ttl)
* twig:
    Twig
* urbiscript:
    UrbiScript (filenames *.u)
```

```
* vala, vapi:
  Vala (filenames *.vala, *.vapi)
* vb.net, vbnet:
  VB.net (filenames *.vb, *.bas)
* vctreestatus:
  VCTreeStatus
* velocity:
  Velocity (filenames *.vm, *.fhtml)
* verilog, v:
  verilog (filenames *.v)
* vgl:
  VGL (filenames *.rpf)
* vhdl:
  vhdl (filenames *.vhdl, *.vhd)
* vim:
  VimL (filenames *.vim, .vimrc, .exrc, .gvimrc, _vimrc, _exrc, _gvimrc, vimrc,
     gvimrc)
* x10, xten:
  X10 (filenames *.x10)
* xml+cheetah, xml+spitfire:
  XML+Cheetah
* xml+django, xml+jinja:
  XML+Django/Jinja
* xml+erb, xml+ruby:
  XML+Ruby
* xml+evoque:
  XML+Evoque (filenames *.xml)
* xml+lasso:
  XML+Lasso
* xml+mako:
  XML+Mako
* xml+myghty:
  XML+Myghty
* xml+php:
  XML+PHP
* xml+smarty:
  XML+Smarty
* xml+velocity:
  XML+Velocity
* xml:
  XML (filenames *.xml, *.xsl, *.rss, *.xslt, *.xsd, *.wsdl, *.wsf)
* xquery, xqy, xq, xql, xqm:
  XQuery (filenames *.xqy, *.xquery, *.xq, *.xql, *.xqm)
* xslt:
  XSLT (filenames *.xsl, *.xslt, *.xpl)
* xtend:
  Xtend (filenames *.xtend)
* xul+mozpreproc:
  XUL+mozpreproc (filenames *.xul.in)
* yaml+jinja, salt, sls:
  YAML+Jinja (filenames *.sls)
```

```
* yaml:
    YAML (filenames *.yaml, *.yml)
* zephir:
    Zephir (filenames *.zep)

Formatters:
~~~~~~~~~~~
* bbcode, bb:
    Format tokens with BBcodes. These formatting codes are used by many bulletin
        boards, so you can highlight your sourcecode with pygments before posting it
        there.
* bmp, bitmap:
    Create a bitmap image from source code. This uses the Python Imaging Library to
        generate a pixmap from the source code. (filenames *.bmp)
* gif:
    Create a GIF image from source code. This uses the Python Imaging Library to
        generate a pixmap from the source code. (filenames *.gif)
* html:
    Format tokens as HTML 4 ‘‘<span>‘‘ tags within a ‘‘<pre>‘‘ tag, wrapped in a
        ‘‘<div>‘‘ tag. The ‘‘<div>‘‘’s CSS class can be set by the ‘cssclass‘
        option. (filenames *.html, *.htm)
* img, IMG, png:
    Create a PNG image from source code. This uses the Python Imaging Library to
        generate a pixmap from the source code. (filenames *.png)
* irc, IRC:
    Format tokens with IRC color sequences
* jpg, jpeg:
    Create a JPEG image from source code. This uses the Python Imaging Library to
        generate a pixmap from the source code. (filenames *.jpg)
* latex, tex:
    Format tokens as LaTeX code. This needs the ‘fancyvrb‘ and ‘color‘ standard
        packages. (filenames *.tex)
* raw, tokens:
    Format tokens as a raw representation for storing token streams. (filenames
        *.raw)
* rtf:
    Format tokens as RTF markup. This formatter automatically outputs full RTF
        documents with color information and other useful stuff. Perfect for Copy
        and Paste into Microsoft(R) Word(R) documents. (filenames *.rtf)
* svg:
    Format tokens as an SVG graphics file. This formatter is still experimental.
        Each line of code is a ‘‘<text>‘‘ element with explicit ‘‘x‘‘ and ‘‘y‘‘
        coordinates containing ‘‘<tspan>‘‘ elements with the individual token
        styles. (filenames *.svg)
* terminal, console:
    Format tokens with ANSI color sequences, for output in a text console. Color
        sequences are terminated at newlines, so that paging the output works
        correctly.
* terminal16m, console16m, 16m:
    Format tokens with ANSI color sequences, for output in a true-color terminal or
        console. Like in ‘TerminalFormatter‘ color sequences are terminated at
```

```
        newlines, so that paging the output works correctly.
* terminal256, console256, 256:
    Format tokens with ANSI color sequences, for output in a 256-color terminal or
        console. Like in ‘TerminalFormatter‘ color sequences are terminated at
        newlines, so that paging the output works correctly.
* testcase:
    Format tokens as appropriate for a new testcase.
* text, null:
    Output the text unchanged without any formatting. (filenames *.txt)

Filters:
~~~~~~~~
* raiseonerror:
    Raise an exception when the lexer generates an error token.
* whitespace:
    Convert tabs, newlines and/or spaces to visible characters.
* tokenmerge:
    Merges consecutive tokens with the same token type in the output stream of a
        lexer.
* highlight:
    Highlight a normal Name (and Name.*) token with a different token type.
* gobble:
    Gobbles source code lines (eats initial characters).
* codetagify:
    Highlight special code tags in comments and docstrings.
* keywordcase:
    Convert keywords to lowercase or uppercase or capitalize them, which means
        first letter uppercase, rest lowercase.

Styles:
~~~~~~~
* manni:
    A colorful style, inspired by the terminal highlighting style.
* igor:
    Pygments version of the official colors for Igor Pro procedures.
* lovelace:
    The style used in Lovelace interactive learning environment. Tries to avoid the
        "angry fruit salad" effect with desaturated and dim colours.
* xcode:
    Style similar to the Xcode default colouring theme.
* vim:
    Styles somewhat like vim 7.0
* autumn:
    A colorful style, inspired by the terminal highlighting style.
* vs:

* rrt:
    Minimalistic "rrt" theme, based on Zap and Emacs defaults.
* native:
    Pygments version of the "native" vim theme.
* perldoc:
```

```
    Style similar to the style used in the perldoc code blocks.
* borland:
    Style similar to the style used in the borland IDEs.
* tango:
    The Crunchy default Style inspired from the color palette from the Tango Icon
        Theme Guidelines.
* emacs:
    The default style (inspired by Emacs 22).
* friendly:
    A modern style based on the VIM pyte theme.
* monokai:
    This style mimics the Monokai color scheme.
* paraiso-dark:

* colorful:
    A colorful style, inspired by CodeRay.
* murphy:
    Murphy's style from CodeRay.
* bw:

* pastie:
    Style similar to the pastie default style.
* algol_nu:

* paraiso-light:

* trac:
    Port of the default trac highlighter design.
* default:
    The default style (inspired by Emacs 22).
* algol:

* fruity:
    Pygments version of the "native" vim theme.
```