## [Ruby文件操作](http://www.cnblogs.com/zhangfei/archive/2012/08/20/2647481.html)

一、新建文件  
    f=File.new(File.join("C:","Test.txt"), "w+")  
    f.puts("I am Jack")  
    f.puts("Hello World")  
  
文件模式  
  
"r" ：Read-only. Starts at beginning of file (default mode).  
  
"r+" ：Read-write. Starts at beginning of file.  
  
"w" ：Write-only. Truncates existing file to zero length or creates a new file for writing.  
  
"w+" ：Read-write. Truncates existing file to zero length or creates a new file for reading and writing.  
  
"a" ：Write-only. Starts at end of file if file exists; otherwise, creates a new file for writing.  
  
"a+" ：Read-write. Starts at end of file if file exists; otherwise, creates a new file for reading and writing.  
  
"b" ：(DOS/Windows only.) Binary file mode. May appear with any of the key letters listed above  
  
二、读取文件  
    file=File.open(File.join("C:","Test.txt"),"r")  
    file.each { |line| print "#{file.lineno}.", line }  
    file.close  
  
三、新建、删除、重命名文件  
    File.new( "books.txt", "w" )  
    File.rename( "books.txt", "chaps.txt" )  
    File.delete( "chaps.txt" )  
  
四、目录操作  
1     创建目录  
    Dir.mkdir("c:/testdir")  
04     #删除目录  
05     Dir.rmdir("c:/testdir")  
07     #查询目录里的文件  
08     p Dir.entries(File.join("C:","Ruby")).join(' ')  
10     #遍历目录  
11     Dir.entries(File.join("C:","Ruby")).each {  
          |e| puts e  
    }  
  
1、ARGV and ARGF  
  
ARGV  
    ARGV << "cnblogslink.txt"  
    #The gets method is a Kernel method that gets lines from ARGV  
    print while gets  
    p ARGV.class  
  
ARGF  
    while line = ARGF.gets  
     print line  
    end  
  
2、文件信息查询  
    #文件是否存在  
    p File::exists?( "cnblogslink.txt" ) # => true  
    #是否是文件  
    p File.file?( "cnblogslink.txt" ) # => true  
    #是否是目录  
    p File::directory?( "c:/ruby" ) # => true  
    p File::directory?( "cnblogslink.txt" ) # => false  
    #文件权限  
    p File.readable?( "cnblogslink.txt" ) # => true  
    p File.writable?( "cnblogslink.txt" ) # => true  
    p File.executable?( "cnblogslink.txt" ) # => false  
    #是否是零长度  
    p File.zero?( "cnblogslink.txt" ) # => false  
    #文件大小 bytes  
    p File.size?( "cnblogslink.txt" ) # => 74  
    p File.size( "cnblogslink.txt" ) # => 74  
    #文件或文件夹  
    p File::ftype( "cnblogslink.txt" ) # => "file"  
    #文件创建、修改、最后一次存取时间  
    p File::ctime( "cnblogslink.txt" ) # => Sat Sep 19 08:05:07 +0800 2009  
    p File::mtime( "cnblogslink.txt" ) # => Sat Sep 19 08:06:34 +0800 2009  
    p File::atime( "cnblogslink.txt" ) # => Sat Sep 19 08:05:07 +0800 2009  
  
3、查找文件  
    puts "查找目录下所有文件及文件夹"   
    Dir["c:/ruby/\*"].each {|x|   
          puts x  
    }   
    puts "条件查询"   
    Dir.foreach('c:/ruby') {   
        |x| puts x if x != "." && x != ".."  
    }  
    puts "查找某一类型文件"  
    Dir["\*.rb"].each {|x|   
      puts x  
     }  
    puts "Open 查询"  
    Dir.open('c:/ruby') { |d| d.grep /l/ }.each{|x| puts x}  
    puts "---------------------------"        
    puts "正则表达式查询"  
    Dir["c:/ruby/ruby/[rs]\*"].each{|x| puts x}   
    puts "------------------------"  
    Dir["c:/ruby/[^s]\*"].each{|x| puts x}  
    puts "------------------------"      
    Dir["c:/ruby/{ruby,li}\*"].each{|x| puts x}   
    puts "------------------------"      
    Dir["c:/ruby/?b\*"].each{|x| puts x}          
    puts "查找目录及子目录的文件"  
    require 'find'       
    Find.find('./') { |path| puts path }

**3、查询目录及子目录文件**  
    require "find"  
Find.find("/etc/passwd", "/var/spool/lp1", ".") do |f|  
  Find.prune if f == "."  
  puts f  
end  
原型：ref.find( [ aName ]\* ) {| aFileName | block }  
prune：Skips the current file or directory, restarting the loop with the next entry. If the current file is a directory, that directory will not be recursively entered. Meaningful only within the block associated with Find::find.  
  
**4、文件比较 复制等**  
    require 'ftools'   
    File.copy 'testfile', 'testfile1'  » true   
    File.compare 'testfile', 'testfile1'  » true