Methods of *Machine* class

- Machine.new(*String* or *CLibIPAddr*: machine address)
  - IP address and FQDN(ex: aaa.bbb.jp) cab be used as machine address
- Machine#set_auth_info(*String*: user name, *String*: password)
  - sets user name and password to access a machine via SSH
- Machine#set_auth_info(*String*: user name, *String*: passphrase of private key , *String*: file path of private key)
  - sets user name, passphrase of private key and file path of private key (ex: ~/.ssh/id_rsa) to access a machine via SSH
- Machine#establish_session()
  - establishes a transport for command execution via SSH
  - before calling this function, authentication information should be set appropriately by Machine#set_auth_info
  - methods described on following slides can be used after calling of this function

Methods of *Machine* class

- Machine#exec!(*String:* command string to execute on remote shell)
  - executes passed command string on remote machine via SSH
  - calling without exclamation mark (!) executes commands asynchronously (no blocking on call)
  - returns standard output of executed command (currently, asynchronous call doesn't return stdout)

- Machine#exec_script_on(*String*: local file path of shell script to execute, *String*: arguments of shell script, *String*: current path which is used at shell script execution)
  - executes passed shell script on remote. you had better pass "." if you don't have any request.
  - returns standard output of executed command as *String.*

- Machine#install_package(*String*: package name to install)
  - Install specified package with package system; yum, apt and so on.
  - package system used by varies platform specified by user. default is yum.
  - returns standard output of package system control command as *String.*

Methods of *Machine* class (File I/O)

- Machine#push_a_file(*String:* local file path, *String*: remote file path)
  - send (copy) a specified local file to remote path
- Machine#pull_a_file(*String*:r emote path, *String*: local path)
  - send (copy) a specified remote file to local path

- Machine#push_files(*String :* l ocal directory path, *String* : remote directory path)
  - send (copy) all files on specified local directory to remote directory
  - subdirectories on the local path are ignored
- Machine#pull_files(*String*: remote direcotry path , *String* : local directory path)
  - send (copy) all files on specified local directory to remote directory
  - subdirectories on the local path are ignored

- Machine#push_dir(*String* : local directory path, *String* : remote directory path)
  - Send (copy) specified local directory to remote path
  - subdirectories are also sent
- Machine#pull_dir(*String* : remote directory path, *String* : local directory path)
  - Send (copy) specified remote directory to localo path
  - subdirectories are also sent

Methods of *Machine* class (config file editing)

- Machine#get_config_file(*String*: file path of configuration path located on remote)
  - return *ConfigFile class instance which represents specified remote config file*

- user can edit remote config file through methods of *ConfigFile* class instance
- editing will reflected to real config file after calling *ConfigFile#save method*

- Methods of ConfigFile class
  - ConfigFile#remove_col_by_str(*String*: string contained by columns you want to remove)
    - remove all columns which contains specified string
  - ConfigFile#replace_col(*String*: string contained by columns you want to replace, *String*: string placed at replaced part)
    - remove all columns which contains specified string
  - ConfigFile#append_str(*String*: string to append)
    - append specified string to end of file
    - multi column (contains new-line character)
  - ConfigFile#save()
    - reflect editing to the *ConfigFile* class instance to remote config file
    - this method has only to be called once after all editing