



```
chefdk# knife cookbook create myvim
chefdk# vi chef-
repo/cookbooks/myvim/recipes/default.rb
execute "apt-get update" do
  command "apt-get update"
end
apt_package "vim" do
  action :install
end
```

```
chefdk# knife cookbook upload myvim  
chefdk# knife node run_list add node1  
  "recipe[myvim]"  
chefnode# chef-client
```

Run the chef-client command ffrom DK:

```
chefdk# knife ssh 'name:node1'  
'sudo chef-client' -i key
```

```
chefdk# knife node show node1
```

```
chefdk# knife node show node1 -l
```

To List in Json format:

```
chefdk# knife node show node1 -Fj
```

To find the hostname of Node:

```
chefdk# knife node show node1 -a fqdn
```

httpd cookbook

knife cookbook create apache

```
chefdk# vi chef-
  repo/cookbooks/apache/recipes/default.rb
execute "yum -y update" do
  command "yum -y update"
end
package "httpd" do
  action :install
end
service "httpd" do
  action [:enable, :start]
end
cookbook_file "/var/www/html/index.html" do
  source "index.html"
  mode "0667"
end
```

```
chefdk# vi /chef-
repo/cookbooks/apache/files/default/index.html
<html>
<body>
<h1> Welcome to chef ! </h1>
</body>
</html>
```

```
chefdk# knife cookbook upload apache
chefdk# knife node run_list add node2
  "recipe[apache]"
chefnode# chef-client
chefnode#cd
  /var/chef/cache/cookbooks/apache/recipes/
```

Chef supports all Ruby loop structures for creating loops inside recipes

```
chefdk # knife cookbook create loops
```

```
chefdk# vi chef-
```

```
repo/cookbooks/loops/recipes/default.rb
```

```
packages = ['vim', 'git', 'curl']
```

```
packages.each do |package|
```

```
    apt_package package do
```

```
        action :install
```

```
    end
```

```
End
```

```
chefdk # knife cookbook create mycond
chefdk# vi chef-
      repo/cookbooks/mycond/recipes/default.rb

if node['platform'] == 'debian' || node['platform']
  == 'ubuntu'
  execute "apt-get update" do
    command "apt-get update"
  end
  apt_package "apache2" do
    action :install
  end
end
```

CASE Example

```
case node['platform']
when 'debian', 'ubuntu'
  execute "apt-get update" do
    command "apt-get update"
  end
  ['openjdk-7-jdk','tomcat7'].each do |pack1|
    apt_package pack1 do
      action :install
    end
  end
end
```

CASE Example

```
when 'redhat', 'centos', 'fedora'  
  execute "yum -y update" do  
    command "yum -y update"  
  end  
  ['tomcat'].each do | pack2 |  
    yum_package pack2 do  
      action :install  
    end  
  end  
  service "tomcat" do  
    action [:enable, :start]  
  end  
end
```

Setting Attributes in Attribute files

Attributes can be set in the cookbook's attributes file

`./cookbooks/<cookbook>/attributes/default.rb`

Format is

`(attribute name) (attribute value)`

`default["apache"]["dir"] = "/etc/apache2"`

They can also be set directly in Recipes

(precedence) (attribute name) (attribute value)

`node.default["apache"]["dir"] = "/etc/apache2"`

An attribute value can be an array

`default["apache"]["listen_ports"] = ["80","443"]`

it can be a hash

- `default["apache"]["site1"] = { "port" => 80 }`
- `default["apache"]["site2"] = { "port" => 81 }`

```
#vi cookbooks/apache/attributes/default.rb
default["apache"]["indexfile"] = "index1.html"
#vi cookbooks/apache/files/default/index1.html
<html>
  <h1>This is index1.html</h1>
</html>
#vi cookbooks/apache/files/default/index2.html
<html>
  <h1>This is index2.html</h1>
</html>
```

```
#vi cookbooks/apache/recipes/default.rb
node.default["apache"]["indexfile"] = "index2.html"
cookbook_file "/var/www/html/index.html" do
  source node["apache"]["indexfile"]
  mode "0644"
end
#knife cookbook upload apache
#knife node run_list add node2 "recipe[apache]"
node# chef-client
(you can check index1.html is will override by index2.html)
```

Attribute Priority

Highest Priority

Defined in a Role

Defined in the Environment

Defined in a Recipe

Lowest Priority

Defined in an Attribute File