

Inheritance

- Classes also support a simple form of object inheritance. For those not acquainted with programming terms, this means that we can extend the functionality of the previous class without copy/pasting the entire class.
- Inheritance allows subclasses to override resource settings defined in parent classes. A class can only inherit from one other class, not more than one.

Ex:

```
class freebsd inherits unix {  
  File['/etc/passwd'] { group => wheel }  
  File['/etc/shadow'] { group => wheel }  
}
```

```
class unix
{
  file{'/tmp/demo.txt':
    ensure => present,
    owner => root,
    group => root,
    content => 'Hello Good day!',
  }
}

class linux inherits unix
{
  File['/tmp/demo.txt']{
    group => puppet,
    mode => 0777,
    content => 'SATHYA TECH HYD',
  }
}

include linux
```

Chaining resources

resources may be chained together to declare relationships between and among them.

```
exec {'update':  
    command => '/usr/bin/apt-get update',  
}  
package {'wget':  
    ensure => installed,  
}  
Exec['update']->Package['wget']
```

Chaining resources

```
exec {'update':  
      command => '/usr/bin/apt-get update',  
      before => Package['wget'],  
}  
  
package {'wget':  
      ensure => installed,  
      require => Exec['update'],  
}
```

Chaining resources

```
package { '.openssh-server':
  ensure => present,
  before => File['/etc/ssh/sshd_config'],
}

file { '/etc/ssh/sshd_config':
  ensure  => file,
  mode    => '0600',
  source  =>
  'puppet:///modules/sshd/sshd_config',
  require => Package['openssh-server'],
}
```

before and require meta-parameter



Chaining resources

```
file { '/etc/ssh/sshd_config':
  ensure => file,
  mode   => '0600',
  source  =>
  'puppet:///modules/sshd/sshd_config',
  notify  => Service['sshd'],
}

service { 'sshd':
  ensure     => running,
  enable     => true,
  subscribe =>
  File['/etc/ssh/sshd_config'],
}
```

notify and subscribe meta-parameter

Arrays

- Puppet allows usage of arrays in various areas.
- Arrays are defined in puppet look like this:
['one', 'two', 'three']

Ex:

```
host { 'one.example.com':  
    host_aliases => ['sathya','tech','hyd'],  
    ip => '192.168.100.1',  
    ensure => present,  
}
```

Variables

- Puppet supports variables like most other languages you may be familiar with. Puppet variables are denoted with \$:

Ex:

```
$content = 'some content \n'  
file { '/tmp/testing':  
    content => $content  
}
```

Conditional Statements

```
if $operatingsystem =='Ubuntu'{  
    package {'apache2':  
        ensure => installed,  
    }  
}  
elsif $operatingsystem == 'CentOS' {  
    package {'httpd':  
        ensure => installed,  
    }  
}
```