



INT-362 - The constructor errors in your `Shop` and `HolidayShop` classes arise because

Cloud Essentials (Lovely Professional University)



Scan to open on Studocu

INT-362 Help Book

Download VMware workstation(Windows) or fusion(MAC)

Steps:

1. Go to broadcom.com
2. Select Support Portal in the upper right corner
3. Log in or register for a Broadcom account
4. Go to support.broadcom.com
5. Select the VMware Cloud Foundation division from the dropdown
6. Click My Downloads on the left
7. Search for Workstation
8. Click the product name
9. Select the Personal Use edition from the dropdown
10. Choose a version and download

Download VirtualBox(for Windows & MAC)

Steps:

1. Go to <https://www.virtualbox.org/wiki/Downloads>
2. Download whichever is applicable

VirtualBox Platform Packages

VirtualBox 7.1.4 platform packages

-  **Windows hosts**
-  **macOS / Intel hosts**
-  **macOS / Apple Silicon hosts**
-  **Linux distributions**
-  **Solaris hosts**
-  **Solaris 11 IPS hosts**

3.

Download ISO file Ubuntu 20.04

Steps:

1. Go to <https://releases.ubuntu.com/focal/ubuntu-20.04.6-desktop-amd64.iso>
2. Download this iso to be used for VM creation in further steps

Creation of a Virtual Machine (VMware workstation)

Steps:

1. Launch VMware Workstation: Open the VMware Workstation application.
2. Start the creation wizard: Go to "File" > "New Virtual Machine".
3. Choose OS type: Select the operating system you want to install on the virtual machine (e.g., Windows, Linux).
4. Name the VM: Provide a name for your virtual machine.
5. Allocate resources: Assign the amount of CPU cores, RAM, and hard disk space you want to allocate to the virtual machine.
6. Create virtual disk: Decide if you want to create a new virtual disk or use an existing one.
7. Network configuration: Set up the network settings for your virtual machine, choosing which network adapter to use and how it will connect to your physical network.
8. Review and finish: Review the configuration details and click "Finish" to create the virtual machine

Creation of a Virtual Machine (VMware fusion)

Steps:

1. Select Workloads > Create VM
2. Enter the VM configuration details, including:
3. VM Location: The name, vCenter, data center, and cluster where you want to create the VM

4. Operating System and Hardware: The required operating system and hardware configuration for the VM
5. Storage: The datastore or cluster on which you want the VM to be created
6. Networking: Select a network or proceed with the default selection

Creation of a Virtual Machine(VirtualBox)

Steps:

1. Launch VirtualBox: Open the VirtualBox application on your computer.
2. Click "New": In the main window, click the "New" button to start creating a new virtual machine.
3. Name your VM: Enter a descriptive name for your virtual machine in the "Name" field.
4. Select Type and Version: Choose the operating system type (e.g., Linux, Windows) and the desired version from the dropdown menu.
5. Allocate Memory: Specify the amount of RAM you want to assign to the virtual machine.
6. Create Virtual Hard Disk: Select "Create a virtual hard disk now" and click "Create".
7. Choose Disk Type: Select "VDI" (VirtualBox Disk Image) as the storage format.
8. Storage Allocation: Decide whether to allocate the disk space dynamically (growing as needed) or to allocate all space at once.
9. Set Disk Size: Specify the total size of the virtual hard disk.
10. Add Storage: Go to the "Settings" of your virtual machine, navigate to the "Storage" tab, and click on the empty CD icon to add an ISO image of the operating system you want to install.

Preparation in Virtual Box for Ubuntu:

Adding your user to sudoers:

Steps:

1. The keyword sudo might needed to be added to sudoers in case of virtualbox

2. Try running command

`sudo apt-get update`

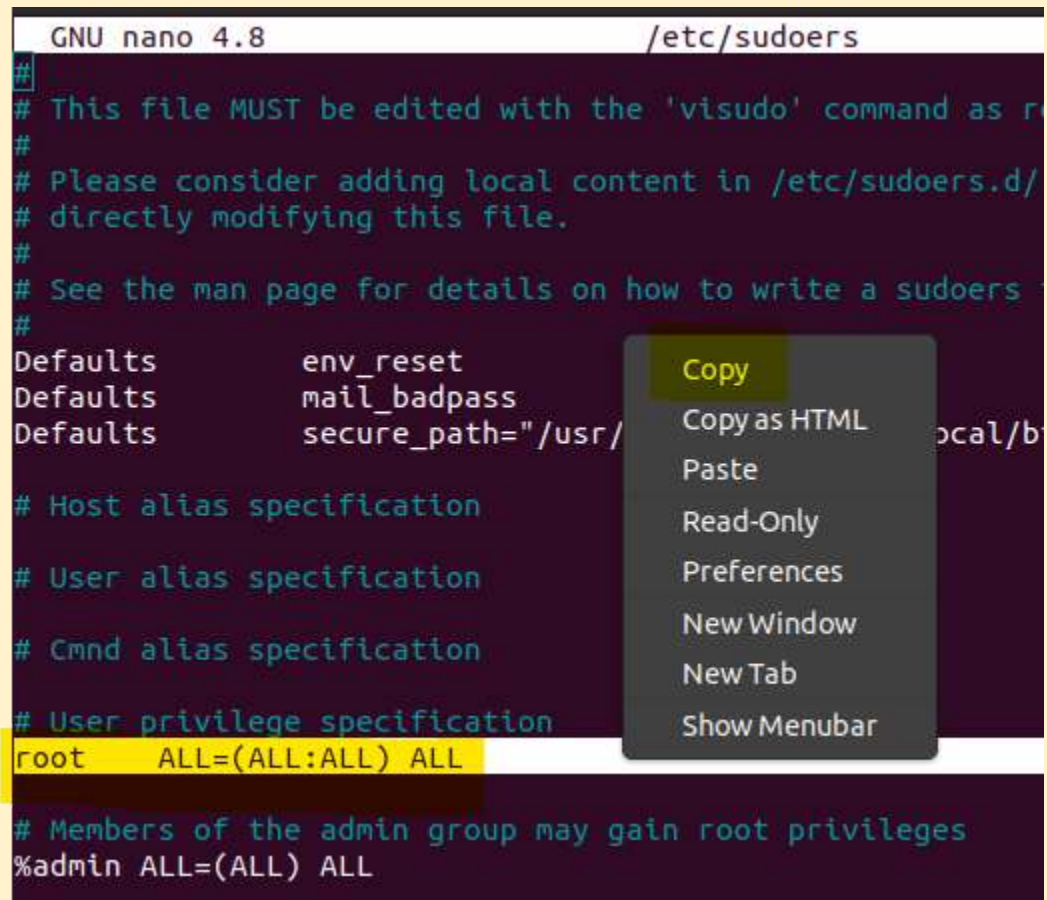
Did it work? No..... Then follow:

```
gunseerat@59u:~$ su root
Password: 
```

This will ask for password, just provide root password

```
gunseerat@59u:~$ su root
Password:
root@59u:/home/gunseerat# nano /etc/sudoers
```

Next open the file in nano editor



```
GNU nano 4.8 /etc/sudoers
#
# This file MUST be edited with the 'visudo' command as root.
#
# Please consider adding local content in /etc/sudoers.d/
# directly modifying this file.
#
# See the man page for details on how to write a sudoers file.
#
Defaults        env_reset
Defaults        mail_badpass
Defaults        secure_path="/usr/local/bin"

# Host alias specification

# User alias specification

# Cmnd alias specification

# User privilege specification
root ALL=(ALL:ALL) ALL

# Members of the admin group may gain root privileges
%admin ALL=(ALL) ALL
```

Copy this line and paste it below in this file anywhere

```
#includedir /etc/sudoers.d
gunseerat ALL=(ALL:ALL) ALL
```

Replace root with your own username

Then save the file with CTRL + S

And exit with CTRL + X

```
root@59u: /home/gunseerat# usermod -aG sudo gunseerat
```

Give this command as

```
usermod -aG sudo [username]
```

Initial prep:

Commands:

1. Update ubuntu

```
sudo apt-get update
```
2. Check networks and interfaces

```
ip a
```
3. Go to root mode

```
su root
```
4. Check Gateways

```
ip route
```

Change Root password (only needed for Workstation/fusion):

Steps:

On terminal write:

```
su root
```

You will enter root mode, Now write command

```
passwd root
```

Give your password twice and you have set root password

Checking and Changing hostname

Steps:

1. Check your IP address, make note of it

`ip a`

2. Run command

`hostname -fqdn`

`sudo nano /etc/hosts`

3. Open file and add to third line, Edit with your ip address and write a domain name of your choice

Here 10.0.2.15 is ip address, apache.c1.a1 is domain name and cloud is nickname

```
root@18u: /home/gunseerat
GNU nano 4.8 /etc/hosts
127.0.0.1 localhost
127.0.1.1 18u.myguest.virtualbox.org
10.0.2.15 apache.c1.a1 cloud
```

4. Save and exit

5. Run command

`sudo hostnamectl set-hostname [nickname]`

Eg: `sudo hostnamectl set-hostname cloud`

6. Check again

Steps to setup network manager- static IP for ubuntu

Check for file in the following path:

1. Computer > etc > netplan

There should be a network-manager-all.yaml file

2. The following needs to be pasted in that file
3. Open terminal

`sudo nano /etc/netplan/01-network-manager-all.yaml`

Paste this into file by erasing previous data: make changes in highlighted areas!!!

```
network:
  version: 2
  renderer: networkd
  ethernets:
    enp1s0:
      dhcp4: no
      dhcp6: no

  bridges:
    cloudbr0:
      interfaces: [enp1s0]
      dhcp4: no
      dhcp6: no
      addresses: [192.168.122.10/24]
      gateway4: 192.168.122.1
      nameservers:
        addresses: [8.8.8.8, 8.8.4.4]
```

The highlighted ones need to be changed with your own addresses and interfaces, use ip a

```
root@cloud:/home/gunseerat# ip a
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group default qlen 1000
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
    inet 127.0.0.1/8 scope host lo
        valid_lft forever preferred_lft forever
    inet6 ::1/128 scope host
        valid_lft forever preferred_lft forever
2: ens33: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel master cloudbr0 state UP group default qlen 1000
    link/ether 00:0c:29:e8:4c:ff brd ff:ff:ff:ff:ff:ff
    altname enp2s1
3: cloudbr0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc noqueue state UP group default qlen 1000
    link/ether 00:0c:29:e8:4c:ff brd ff:ff:ff:ff:ff:ff
    inet 192.168.189.128/24 brd 192.168.189.255 scope global cloudbr0
        valid_lft forever preferred_lft forever
    inet6 fe80::20c:29ff:fee8:4cff/64 scope link
        valid_lft forever preferred_lft forever
```

and ip r


```
root@cloud:/home/gunseerat# ip r
default via 192.168.189.2 dev cloudbri0 proto static
192.168.189.0/24 dev cloudbri0 proto kernel scope link src 192.168.189.128
root@cloud:/home/gunseerat#
```

Edited file should look like this

```
GNU nano 4.8 /etc/netplan/01-network-manager-all.yaml
network:
  version: 2
  renderer: networkd
  ethernets:
    ens33:
      dhcp4: no
      dhcp6: no

  bridges:
    cloudbri0:
      interfaces: [ens33]
      dhcp4: no
      dhcp6: no
      addresses: [192.168.189.128/24]
      gateway4: 192.168.189.2
      nameservers:
        addresses: [172.19.2.254, 8.8.8.8, 8.8.4.4]
```

After editing the file write the following commands in terminal

```
sudo netplan generate
```

```
sudo netplan apply or sudo netplan -debbbug apply
```

```
sudo systemctl restart NetworkManager
```

Further installations: [ntp, bridge-utils and firewall settings]

Steps:

In terminal type the following commands in order----

```
sudo apt-get install ntp bridge-utils openssh-server -y
```

```
sudo systemctl enable ntp
```

```
sudo systemctl start ntp
```

```
sudo sed -i "s/#PermitRootLogin.*/PermitRootLogin yes/" /etc/ssh/sshd_config
```

```
sudo ufw disable
```

```
root@cloud:/home/gunseerat# sudo apt-get install ntp bridge-utils openssh-server
-y
Reading package lists... Done
Building dependency tree
Reading state information... Done
bridge-utils is already the newest version (1.6-2ubuntu1).
openssh-server is already the newest version (1:8.2p1-4ubuntu0.11).
ntp is already the newest version (1:4.2.8p12+dfsg-3ubuntu4.20.04.1).
0 upgraded, 0 newly installed, 0 to remove and 307 not upgraded.
root@cloud:/home/gunseerat# sudo systemctl enable ntp
Synchronizing state of ntp.service with SysV service script with /lib/systemd/sy
stemd-sysv-install.
Executing: /lib/systemd/systemd-sysv-install enable ntp
root@cloud:/home/gunseerat# sudo systemctl start ntp
root@cloud:/home/gunseerat# sudo sed -i "s/#PermitRootLogin.*/PermitRootLogin ye
s/" /etc/ssh/sshd_config
root@cloud:/home/gunseerat# sudo ufw disable
Firewall stopped and disabled on system startup
root@cloud:/home/gunseerat#
```

Adding Package to apt repository in source list

Steps:

Open file in nano editor

```
sudo nano /etc/apt/sources.list.d/cloudstack.list
```

After opening file add this line to it

```
deb https://download.cloudstack.org/ubuntu focal 4.20
```

Paste it like this

A screenshot of the nano text editor. The title bar at the top reads "GNU nano 4.8 /etc/apt/sources.list.d/cloudstack.list". The main editing area shows a single line of text: "deb https://download.cloudstack.org/ubuntu focal 4.20". The text is color-coded: "deb" is green, the URL is green, "ubuntu" is red, "focal" is red, and "4.20" is purple. The background of the editor is dark purple.

```
GNU nano 4.8 /etc/apt/sources.list.d/cloudstack.list  
deb https://download.cloudstack.org/ubuntu focal 4.20
```

After this run the command

```
wget -O - https://download.cloudstack.org/release.asc |sudo tee  
/etc/apt/trusted.gpg.d/cloudstack.asc
```

```
root@cloud: /home/gunseerat
SLA2TmUQBxnhIxsUw0kto8agTzsgNKG+CE0AaXpohxgVv040ZRoBz+5aZe3XDELR
edjsyVBv7bJd2m9DAVdADjv3JSdlJgntkTE/c1V5GJrtECSkZ3jmAraA6bX8+jWu
BQD+Ym5iRtYydsdN1P09C/qnhf00eTkYcd4wkII6CztCC0ndTX3c2d5e0oQwZsqp
1NUTU9N7nHALx3flIXBqRMBCA8Xa7AE4oCqG8HeY0C3In/LofoemgazEhwARAQAB
tC1BcGFjaGUgQ2xvdWRTdGFjayA8ZGV2QGNsb3Vkc3RhY2suYXBhY2hlLm9yZz6J
AjjgEEwECACIFAlYVI1kCGy8GCwkIBwMCBhUIAgkKCwQWAgMBAh4BAheAAAoJED1i
uDfxA0dY+Nsp/37BRsvx+uxc8NoA88BQ20l6sWrHZ5AoQA30PnV/SUJ8nuEETJ4b
Pp3+vuT2hWTEV6qQX0pirtCbRkFG5626j1P4/F3sDJTtHo0Te0Kd0cI/mUw4LHNH
bunh6WrfLyOWJ0bDrGuso/87kZK1e6SNwD6YxthCTpAX0Ziq5INzsA+ViP7F5U/N
2mXRRcKThIWktyQxmI/jp3MFFmSLg2ds8++HWLcKRp91JHn3xwSZxARLuuiqPRaS
ER2Hmdh30y/bleQn0ZN/MAEgBgid2YfKta58IrUPTibI7LFg9G60iEosnQfuY+Ez
jj2Q1KGPBIADQFzFAsGXMU8PBWuap+3UN6jqlwNLXmKbv4mSic0NRoNhooQWSX1G
uTACBcW9NjGysWaKMP0Wx6LSyJ+cmgnm0k+v1U6mgSPQr1P36pWSAbdSdQR0TnHM
qwe2xBm2DgNroiIfoaUKKh+VNnDXSPp/Ldua4Fk6vZVLYEIGSrUXmGDu/7LJuE9
oez2/b0xJ38pwvX0+cTxxdiHmn37Km20Hwiq03hmryiek70YvqPPlyW+YrKEefsS
LQosKiELe3X2kl5AdNxJC+S5V2RD3Qp5PwDGGpb9VN7IITxGc0w30kgzr9qNeP8e
uknsiiyr0jMXN0TSPWoRnJD85LI13xlSng1ELUhtV09XqP62XNrE3Jmj
=ORlq
-----END PGP PUBLIC KEY BLOCK-----
-
100%[=====>] 1.61K ---KB/s in 0s
2025-02-05 23:42:35 (104 MB/s) - written to stdout [1649/1649]
```

Then run

```
sudo apt update
```

Then install cloudstack as

```
sudo apt install cloudstack-management -y
```

Installation of MySQL server

Steps:

Install mysql server with command

```
sudo apt install mysql-server
```

Look in your files for `/etc/my.cnf` or `/etc/mysql/my.cnf`

Open file in nano editor and add to the file you opened

```
[mysqld]
```

```
server-id=01
```

```
innodb_rollback_on_timeout=1
```

```
innodb_lock_wait_timeout=600
```

```
max_connections=350
```

```
log-bin=mysql-bin
```

```
binlog-format = 'ROW'
```

Create second file on editor as

```
sudo nano /etc/mysql/conf.d/cloudstack.cnf
```

Add to the file

```
[mysqld]
```

```
server-id=01
```

```
innodb_rollback_on_timeout=1
```

```
innodb_lock_wait_timeout=600
```

```
max_connections=350
```

```
log-bin=mysql-bin
```

```
binlog-format = 'ROW'
```

Restart mysql

```
sudo systemctl restart mysql
```

After restarting mysql, run this command by replacing root password with a password of your choice

```
sudo cloudstack-setup-databases cloud:<root-password>@localhost  
--deploy-as=root
```

After running the previous command, run this command and follow up on questions, type y for any questions and type 0 for any password policies.

```
sudo mysql_secure_installation
```

After completion of the previous command run

```
cloudstack-setup-management
```

Setup of NFS shares:

```
apt install nfs-kernel-server
```

Create the /export/primary and /export/secondary directories for the NFS share.

```
sudo mkdir -p /export/primary /export/secondary
```

Configure the new directories as NFS exports.

```
sudo echo "/export *(rw,async,no_root_squash,no_subtree_check)" | sudo tee -a /etc/exports
```

Export the /export directory.

```
sudo exportfs -a
```

Restart the nfs-kernel-server.

```
sudo service nfs-kernel-server restart
```

Create the directories to mount the storages.

```
sudo mkdir -p /mnt/primary /mnt/secondary
```

```
root@cloud:/home/gunseerat# mkdir -p /export/primary /export/secondary
root@cloud:/home/gunseerat# echo "/export *(rw,async,no_root_squash,no_subtree_c
heck)" | sudo tee -a /etc/exports
/export *(rw,async,no_root_squash,no_subtree_check)
root@cloud:/home/gunseerat# exportfs -a
root@cloud:/home/gunseerat# service nfs-kernel-server restart
root@cloud:/home/gunseerat# mkdir -p /mnt/primary /mnt/secondary
root@cloud:/home/gunseerat#
```

Run the commands below to add the export rules in the /etc/fstab file in order to make the mount automatic on system reboot.

Note: Replace the IP addresses according to your network configuration. Give your IP at the highlighted place

```
sudo echo "192.168.122.10:/export/primary /mnt/primary nfs  
rsize=8192,wsiz=8192,timeo=14,intr,vers=3,noauto 0 2" | sudo tee -a /etc/fstab
```

```
sudo echo "192.168.122.10:/export/secondary /mnt/secondary nfs  
rsize=8192,wsiz=8192,timeo=14,intr,vers=3,noauto 0 2" | sudo tee -a /etc/fstab
```

Mount the storages.

```
sudo mount /mnt/primary
```

```
sudo mount /mnt/secondary
```

Setting up Quota

```
apt install quota
```

```
sed -i -e 's/^RPCMOUNTDOPTS="--manage-gids"/RPCMOUNTDOPTS="-p  
892 --manage-gids"/g' /etc/default/nfs-kernel-server
```

```
sed -i -e 's/^STATDOPTS=$/STATDOPTS="--port 662 --outgoing-port 2020"/g'  
/etc/default/nfs-common
```

```
echo "NEED_STATD=yes" >> /etc/default/nfs-common
```

```
sed -i -e 's/^RPCRQUOTADOPTS=$/RPCRQUOTADOPTS="-p 875"/g'  
/etc/default/quota
```

Install cloudstack agent

```
sudo apt install cloudstack-agent -y
```

```
sed -i -e 's/\#vnc_listen.*$/vnc_listen = "0.0.0.0"/g' /etc/libvirt/qemu.conf
```

```
systemctl mask libvirtd.socket libvirtd-ro.socket libvirtd-admin.socket
```

```
libvirtd-tls.socket libvirtd-tcp.socket
```

```
systemctl restart libvirtd
```

```
echo 'listen_tls=0' >> /etc/libvirt/libvirtd.conf
```

```
echo 'listen_tcp=1' >> /etc/libvirt/libvirtd.conf
```

```
echo 'tcp_port = "16509"' >> /etc/libvirt/libvirtd.conf
```

```
echo 'mdns_adv = 0' >> /etc/libvirt/libvirtd.conf
```

```
echo 'auth_tcp = "none"' >> /etc/libvirt/libvirtd.conf
```

```
systemctl restart libvirtd
```

Steps to setup agent

```
apt-get install uuid
```

```
UUID=$(uuid)
```

```
echo host_uuid = \"$UUID\" >> /etc/libvirt/libvirtd.conf
```

```
systemctl restart libvirtd
```

```
ln -s /etc/apparmor.d/usr.sbin.libvirtd /etc/apparmor.d/disable/
```

```
ln -s /etc/apparmor.d/usr.lib.libvirt.virt-aa-helper /etc/apparmor.d/disable/
```

```
apparmor_parser -R /etc/apparmor.d/usr.sbin.libvirtd
```

```
apparmor_parser -R /etc/apparmor.d/usr.lib.libvirt.virt-aa-helper
```


System VM template

```
sudo /usr/share/cloudstack-common/scripts/storage/secondary/cloud-install-sys-tmpl  
-m /mnt/secondary -u  
http://cloudstack.appt-get.eu/systemvm/4.18/systemvmtemplate-4.18.0-kvm.qcow2.bz  
2 -h kvm -F
```

Commands for troubleshooting:

```
journalctl -xeu {service name}
```

```
systemctl status {service name}
```

```
systemctl restart {service name}
```

```
tail -n 50 -f /var/log/cloudstack/management/management-server.log
```

```
grep ERROR /var/log/cloudstack/management/management-server.log
```

