eval array=($(xe vm-list params=uuid is-control-domain=false|grep “uuid”|cut –c 17-))

data=()

data1=()

idx=0

idx1=0

for i in “${array[@]}”

do

vm\_name=$(xe vm-list params=name-label uuid=$i|grep “name-label”|cut –c 23-)

data[idx]=$vm\_name

idx=`expr $idx + 1`

echo “CPU Usage : ${data1[$i]}”

done

l=${#data1[@]}

echo “The CPU values of VMs are :”

for (( i = 0 ; i < l ; i++ ))

do

echo ${data1[$i]}

arr[i]=${data1[i]%%.\*}

done

echo “Sorted Order : ”

for (( i = 0 ; i < l ; i++ ))

do

for (( j = 0 ; j < l ; j++ ))

do

if [ ${arr[$i]} –gt ${arr[$j]} ] ; then

t=${arr[$i]}

arr[$i]=${arr[$j]}

arr[$j]=$t

fi

done

echo ${arr[$i]}

done

if [ `expr $l % 2` -eq 0 ]

then

mid=`expr $l / 2` -eq 0`

echo “Middle Value : $mid”

else

mid=`expr $l / 2`

mid=`expr $mid + 1`

echo “Middle Value : $mid”

fi

echo “array1 : ”

for (( i = 0 ; i < l ; i++ ))

do

arr1[$i]=${arr[$i]}

echo “${arr1[$i]}”

done

echo “array2 :”

j=0

for (( i=$mid ; i < l ; i++ ))

do

arr2[$j]=${arr[$i]}

echo ${arr2[$j]}

j=`expr $j + 1`

done

l1=${#arr1[@]}

l2=${#arr2[@]}

a=`expr $l1 / 2`

b=`expr $l2 / 2`

if [ `expr $l1 % 2` -eq 0 ]

then

mid1=$a

mid2=$b

else

mid1=`expr $a + 1`

mid2=`expr $b + 1`

fi

c=`expr $mid1 - 1`

d=`expr $mid2 – 1`

q1=${arr1[$c]}

q3=${arr2[$d]}

echo “Q1 = $q1 and Q3 = $q3”

IQR=`expr $q3 – $q1`

echo “IQR Value = $IQR”

e=’echo $IQR \\* 1.5|bc’

echo $e

t=$( bc <<< “$e – 1” )

echo “Upper Threshold = $t”

y=${t%%.\*}

for (( i=0 ; i < l; i++ ))

do

if [ `expr ${arr[$i]}` -gt $y ]

then

k=${data[$i]}

echo “Overloaded VM : $k”

xe vm-migrate vm=$k host=xenserver123 –live –force remote-username=root remote-master=172.17.17.211 remote-password=mahatce

else

echo “ Non of the VM is overloaded”

fi

done