Practical No: 1 – K means clustering.
Aim:
Read a datafile grades_km_input.csv and apply k-means clustering.

Datafile:
https://github.com/Mounaki/Clustering/blob/master/grades_km_input.csv

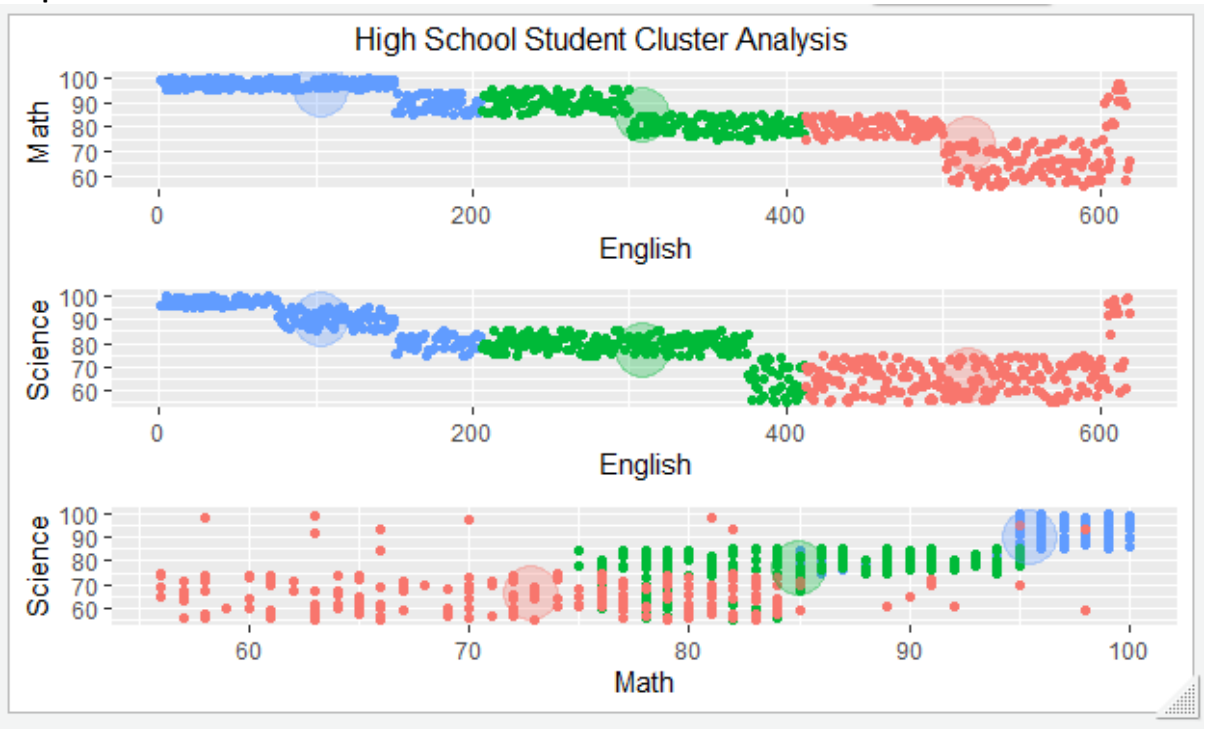
source code:

```
# install required packages

install.packages("plyr")
install.packages("ggplot2")
install.packages("cluster")
install.packages("lattice")
install.packages("grid")
install.packages("gridExtra")

# Load the package

library(plyr)
library(ggplot2)
library(cluster)
library(lattice)
library(grid)
library(gridExtra)

# A data frame is a two-dimensional array-like structure in which each column contains values of one variable and each row contains one set of values from each column.

grade_input=as.data.frame(read.csv("D:/2020/Big Data Analytics/Practical/grades_km_input.csv"))
kmdata_orig=as.matrix(grade_input[, c ("Student","English","Math","Science")])
kmdata=kmdata_orig[,2:4]
kmdata[1:10,]

# the k-means algorithm is used to identify clusters for k = 1, 2, .. . , 15. For each value of k, the WSS is calculated.

wss=numeric(15)

# the option n start=25 specifies that the k-means algorithm will be repeated 25 times, each starting with k random initial centroids

for(k in 1:15)wss[k]=sum(kmeans(kmdata,centers=k,nstart=25)$withinss)

plot(1:15,wss,type="b",xlab="Number of Clusters",ylab="Within sum of square")

#As can be seen, the WSS is greatly reduced when k increases from one to two. Another substantial reduction in WSS occurs at k = 3. However, the improvement in WSS is fairly linear fork > 3.

km = kmeans(kmdata,3,nstart=25)
km
```

Compiled by: Ms. Beena Kapadia
Vidyalankar School of Information Technology

```
c( wss[3] , sum(km$withinss))
df=as.data.frame(kmdata_orig[,2:4])
df$cluster=factor(km$cluster)
centers=as.data.frame(km$centers)
g1=ggplot(data=df, aes(x=English, y=Math, color=cluster )) +
  geom_point() + theme(legend.position="right") +
  geom_point(data=centers,aes(x=English,y=Math, color=as.factor(c(1,2,3))),size=10, alpha=.3,
show.legend =FALSE)
g2=ggplot(data=df, aes(x=English, y=Science, color=cluster )) +
  geom_point () +geom_point(data=centers,aes(x=English,y=Science,
color=as.factor(c(1,2,3))),size=10, alpha=.3, show.legend=FALSE)
g3 = ggplot(data=df, aes(x=Math, y=Science, color=cluster )) +
  geom_point () + geom_point(data=centers,aes(x=Math,y=Science,
color=as.factor(c(1,2,3))),size=10, alpha=.3, show.legend=FALSE)
tmp=ggplot_gtable(ggplot_build(g1))

grid.arrange(arrangeGrob(g1 + theme(legend.position="none"),g2 +
theme(legend.position="none"),g3 + theme(legend.position="none"),top ="High School Student
Cluster Analysis" ,ncol=1))
```

output



Practical no 2: Apriori algorithm

Aim: Perform Apriori algorithm using Groceries dataset from the R arules package.

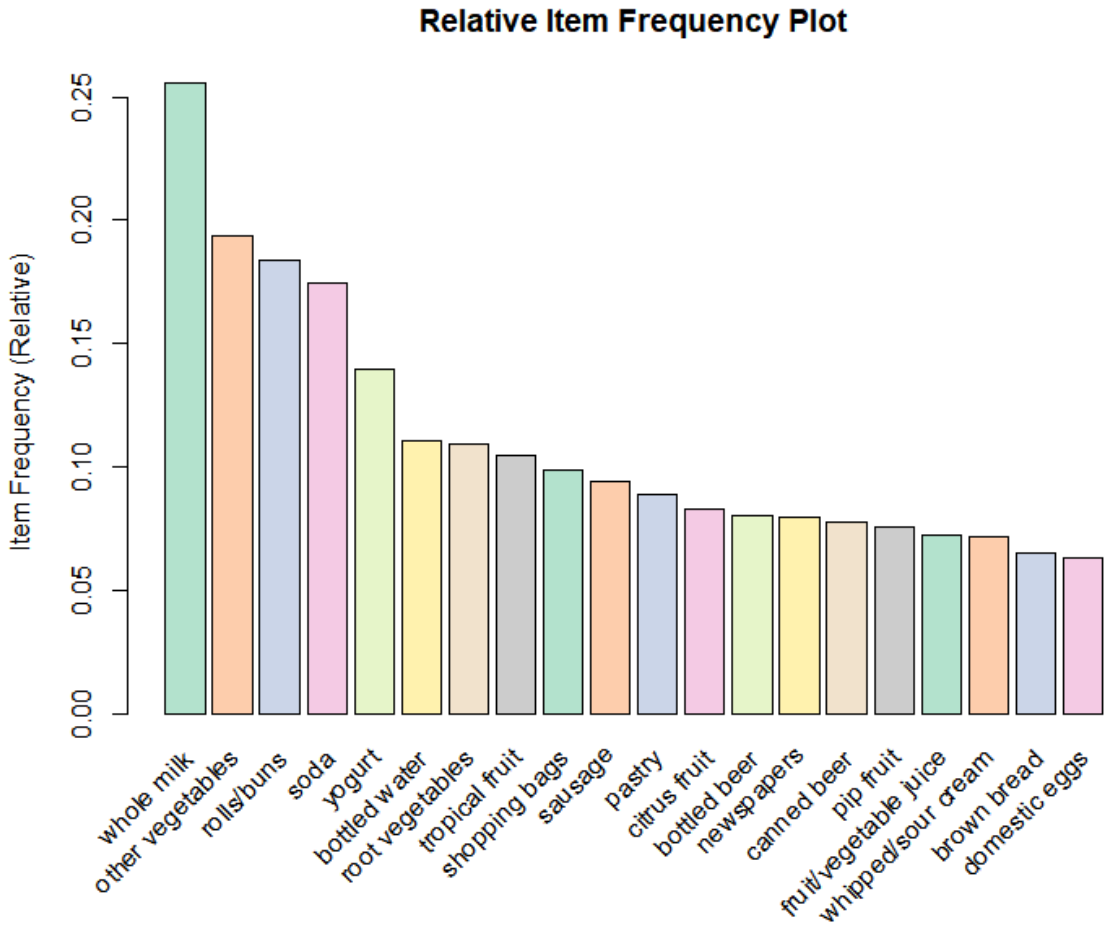
Code:

```
install.packages("arules")
install.packages("arulesViz")
install.packages("RColorBrewer")


# Loading Libraries
library(arules)
library(arulesViz)
library(RColorBrewer)

# import dataset
data(Groceries)
Groceries

summary(Groceries)
class(Groceries)


# using apriori() function
rules = apriori(Groceries, parameter = list(supp = 0.02, conf = 0.2))
summary (rules)

# using inspect() function
inspect(rules[1:10])


# using itemFrequencyPlot() function
arules::itemFrequencyPlot(Groceries, topN = 20,
                  col = brewer.pal(8, 'Pastel2'),
                  main = 'Relative Item Frequency Plot',
                  type = "relative",
                  ylab = "Item Frequency (Relative)")

itemsets = apriori(Groceries, parameter = list(minlen=2, maxlen=2,support=0.02, target="frequent
itemsets"))
summary(itemsets)

# using inspect() function
inspect(itemsets[1:10])

itemsets_3 = apriori(Groceries, parameter = list(minlen=3, maxlen=3,support=0.02, target="frequent
itemsets"))
summary(itemsets_3)

# using inspect() function
```

inspect(itemsets_3)

output:

lhs	rhs	support	confidence	coverage	lift	count
[1] {}	=> {whole milk}	0.25551601	0.2555160	1.00000000	1.000000	2513
[2] {hard cheese}	=> {whole milk}	0.01006609	0.4107884	0.02450432	1.607682	99
[3] {butter milk}	=> {other vegetables}	0.01037112	0.3709091	0.02796136	1.916916	102
[4] {butter milk}	=> {whole milk}	0.01159126	0.4145455	0.02796136	1.622385	114
[5] {ham}	=> {whole milk}	0.01148958	0.4414062	0.02602949	1.727509	113
[6] {sliced cheese}	=> {whole milk}	0.01077783	0.4398340	0.02450432	1.721356	106
[7] {oil}	=> {whole milk}	0.01128622	0.4021739	0.02806304	1.573968	111
[8] {onions}	=> {other vegetables}	0.01423488	0.4590164	0.03101169	2.372268	140
[9] {onions}	=> {whole milk}	0.01209964	0.3901639	0.03101169	1.526965	119
[10] {berries}	=> {yogurt}	0.01057448	0.3180428	0.03324860	2.279848	104



Practical no: 3 Linear regression

Practical no: 3 a) Simple Linear regression

Aim: Create your own data for years of experience and salary in lakhs and apply linear regression model to predict the salary.

Code:

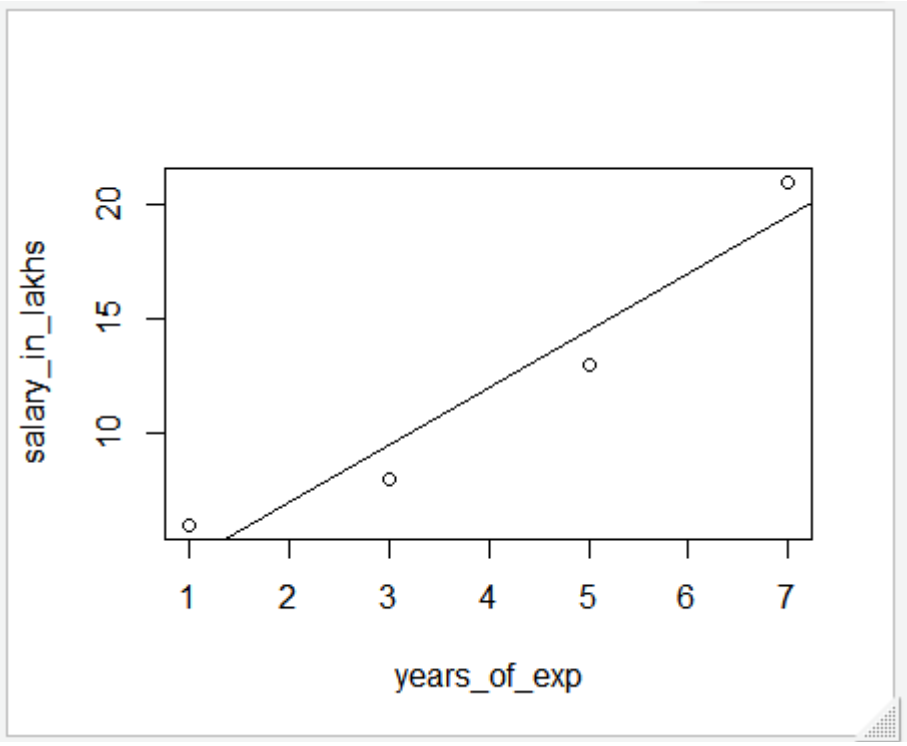
```
years_of_exp = c(7,5,1,3)
salary_in_lakhs = c(21,13,6,8)

#employee.data = data.frame(satisfaction_score, years_of_exp, salary_in_lakhs)
employee.data = data.frame(years_of_exp, salary_in_lakhs)
employee.data

# Estimation of the salary of an employee, based on his year of experience and satisfaction score in his company.

model <- lm(salary_in_lakhs ~  years_of_exp, data = employee.data)
summary(model)

# The formula of Regression becomes
# Y = 2 + 2.5*year_of_Exp

# Visualization of Regression

plot(salary_in_lakhs ~ years_of_exp, data = employee.data)
abline(model)
```

output:

```
years_of_exp salary_in_lakhs
1            7             21
2            5             13
3            1              6
4            3              8

Residuals:
  1   2   3   4
 1.5 -1.5  1.5 -1.5

Coefficients:
             Estimate Std. Error t value Pr(>|t|)
(Intercept)    2.0000     2.1737    0.92   0.4547
years_of_exp   2.5000     0.4743    5.27   0.0342 *
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 2.121 on 2 degrees of freedom
Multiple R-squared:  0.9328,      Adjusted R-squared:  0.8993
```

Compiled by: Ms. Beena Kapadia
Vidyalankar School of Information Technology

F-statistic: 27.78 on 1 and 2 DF, p-value: 0.03417



b) Logistic regression

Aim: Take the in-built data from ISLR package and apply generalized logistic regression to find whether a person would be defaulter or not; considering input as student, income and balance.

Source code:

```
install.packages("ISLR")
library(ISLR)

#load dataset
data <- ISLR::Default
print (head(ISLR::Default))

#view summary of dataset
summary(data)

#find total observations in dataset
nrow(data)

#Create Training and Test Samples
#split the dataset into a training set to train the model on and a testing set to test the model
set.seed(1)

#Use 70% of dataset as training set and remaining 30% as testing set
sample <- sample(c(TRUE, FALSE), nrow(data), replace=TRUE, prob=c(0.7,0.3))
print (sample)

train <- data[sample, ]
test <- data[!sample, ]

nrow(train)
nrow(test)

# Fit the Logistic Regression Model
# use the glm (general linear model) function and specify family="binomial"
#so that R fits a logistic regression model to the dataset

model <- glm(default~student+balance+income, family="binomial", data=train)

#view model summary
summary(model)

#Model Diagnostics
install.packages("InformationValue")
library(InformationValue)
predicted <- predict(model, test, type="response")

confusionMatrix(test$default, predicted)
```

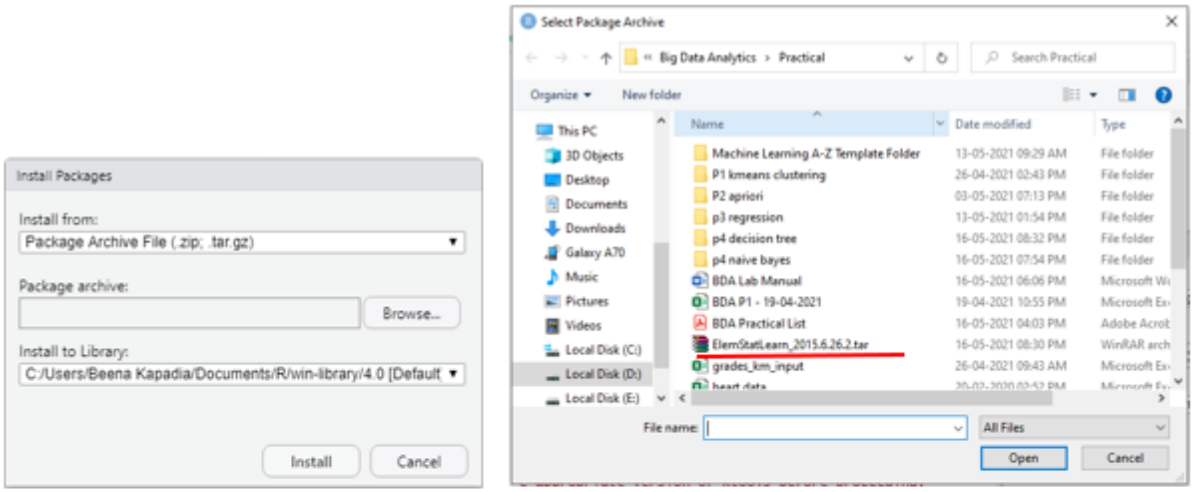
output:

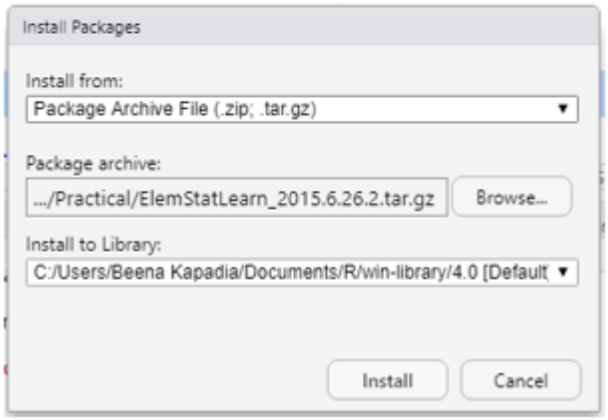
```
> print (head(ISLR::Default))
  default student   balance    income
1      No      No  729.5265 44361.625
2      No     Yes  817.1804 12106.135
3      No      No 1073.5492 31767.139
4      No      No  529.2506 35704.494
5      No      No  785.6559 38463.496
6      No     Yes  919.5885  7491.559

summary(data)
default    student       balance        income
 No :9667   No :7056   Min.   :   0.0   Min.   :  772
 Yes: 333   Yes:2944   1st Qu.: 481.7   1st Qu.:21340
                       Median : 823.6   Median :34553
                       Mean   : 835.4   Mean   :33517
                       3rd Qu.:1166.3   3rd Qu.:43808
                       Max.   :2654.3   Max.   :73554
> nrow(data)
[1] 10000
> print (sample)
  [1]  TRUE  TRUE  TRUE FALSE  TRUE FALSE FALSE  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE
FALSE  TRUE FALSE FALSE
 [19]  TRUE FALSE FALSE  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE FALSE  TRUE  TRUE  TRUE
TRUE  TRUE FALSE  TRUE
> nrow(train)
[1] 6964
> nrow(test)
[1] 3036
> summary(model)

Call:
glm(formula = default ~ student + balance + income, family = "binomial",
    data = train)

Deviance Residuals:
    Min       1Q   Median       3Q      Max
-2.5586  -0.1353  -0.0519  -0.0177   3.7973

Coefficients:
                Estimate    Std. Error z value            Pr(>|z|)
(Intercept) -11.478101194   0.623409555 -18.412 <0.0000000000000002 ***
studentYes   -0.493292438   0.285735949  -1.726              0.0843 .
balance       0.005988059   0.000293765  20.384 <0.0000000000000002 ***
income        0.000007857   0.000009965   0.788              0.4304
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 2021.1  on 6963  degrees of freedom
Residual deviance: 1065.4  on 6960  degrees of freedom
AIC: 1073.4

Number of Fisher Scoring iterations: 8
> confusionMatrix(test$default, predicted)
     0  1
0 2912 64
1   21 39
```

Compiled by: Ms. Beena Kapadia
Vidyalankar School of Information Technology

Practical 4 a Decision Tree

Get ElemStatLearn package from https://cran.r-project.org/src/contrib/Archive/ElemStatLearn/ as shown below:





Code:

```
# Decision Tree Classification

# Importing the dataset
dataset = read.csv('D:\\2020\\Big Data Analytics\\Practical\\p4 decision
tree\\Social_Network_Ads.csv')
dataset = dataset[3:5]

# Encoding the target feature as factor
dataset$Purchased = factor(dataset$Purchased, levels = c(0, 1))

# Splitting the dataset into the Training set and Test set
install.packages('caTools')
library(caTools)
set.seed(123)
split = sample.split(dataset$Purchased, SplitRatio = 0.75)
training_set = subset(dataset, split == TRUE)
```

Compiled by: Ms. Beena Kapadia
Vidyalankar School of Information Technology

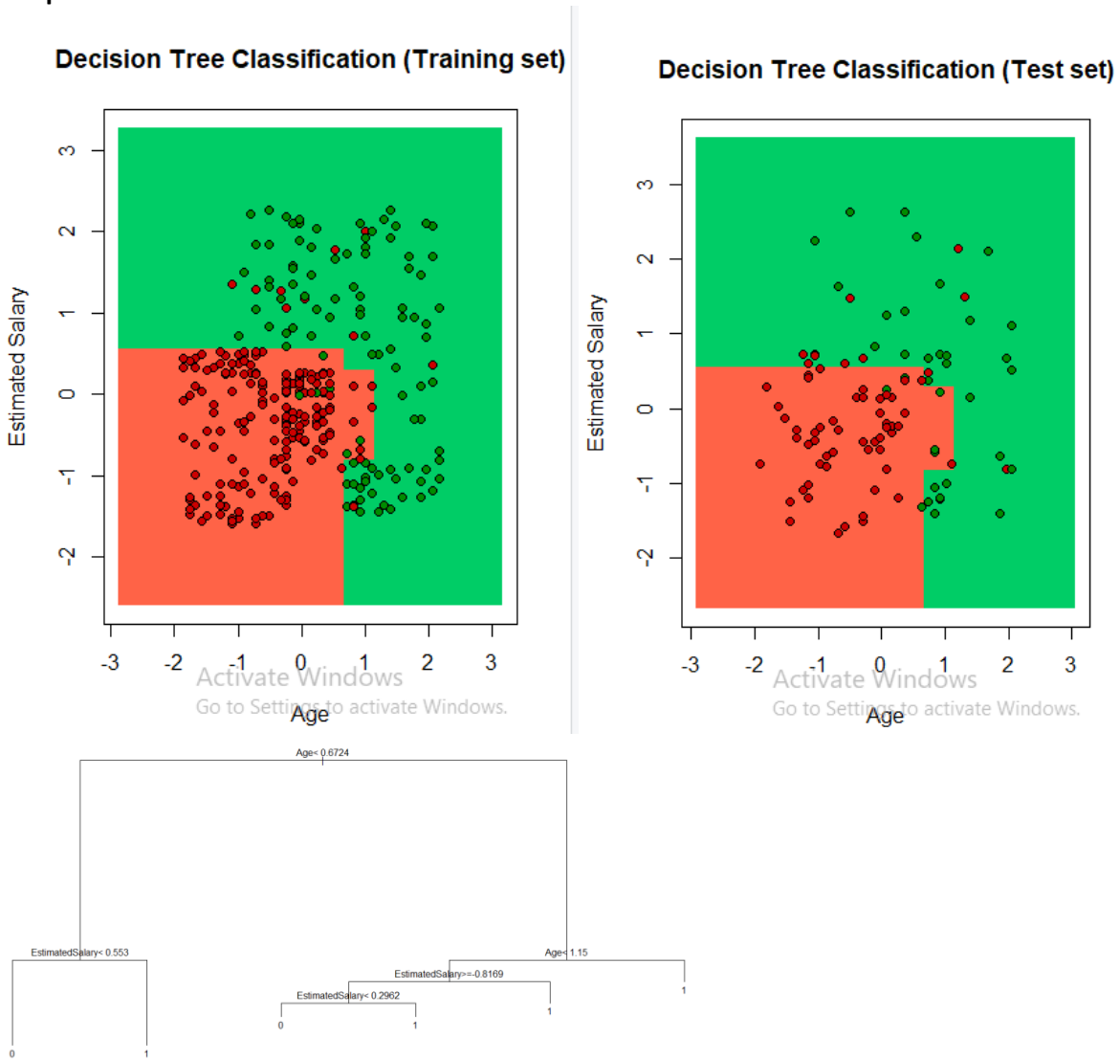
```
test_set = subset(dataset, split == FALSE)

# Feature Scaling
training_set[-3] = scale(training_set[-3])
test_set[-3] = scale(test_set[-3])

# Fitting Decision Tree Classification to the Training set
install.packages('rpart')
library(rpart)
classifier = rpart(formula = Purchased ~ .,
                   data = training_set)

# Predicting the Test set results
y_pred = predict(classifier, newdata = test_set[-3], type = 'class')

# Making the Confusion Matrix
cm = table(test_set[, 3], y_pred)

# Visualising the Training set results
install.packages("ElemStatLearn")
library(ElemStatLearn)
set = training_set
X1 = seq(min(set[, 1]) - 1, max(set[, 1]) + 1, by = 0.01)
X2 = seq(min(set[, 2]) - 1, max(set[, 2]) + 1, by = 0.01)
grid_set = expand.grid(X1, X2)
colnames(grid_set) = c('Age', 'EstimatedSalary')
y_grid = predict(classifier, newdata = grid_set, type = 'class')
plot(set[, -3],
     main = 'Decision Tree Classification (Training set)',
     xlab = 'Age', ylab = 'Estimated Salary',
     xlim = range(X1), ylim = range(X2))
contour(X1, X2, matrix(as.numeric(y_grid), length(X1), length(X2)), add = TRUE)
points(grid_set, pch = '.', col = ifelse(y_grid == 1, 'springgreen3', 'tomato'))
points(set, pch = 21, bg = ifelse(set[, 3] == 1, 'green4', 'red3'))

# Visualising the Test set results
library(ElemStatLearn)
set = test_set
X1 = seq(min(set[, 1]) - 1, max(set[, 1]) + 1, by = 0.01)
X2 = seq(min(set[, 2]) - 1, max(set[, 2]) + 1, by = 0.01)
grid_set = expand.grid(X1, X2)
colnames(grid_set) = c('Age', 'EstimatedSalary')
y_grid = predict(classifier, newdata = grid_set, type = 'class')
plot(set[, -3], main = 'Decision Tree Classification (Test set)',
     xlab = 'Age', ylab = 'Estimated Salary',
     xlim = range(X1), ylim = range(X2))
contour(X1, X2, matrix(as.numeric(y_grid), length(X1), length(X2)), add = TRUE)
points(grid_set, pch = '.', col = ifelse(y_grid == 1, 'springgreen3', 'tomato'))
points(set, pch = 21, bg = ifelse(set[, 3] == 1, 'green4', 'red3'))

# Plotting the tree
```

plot(classifier)
text(classifier)

input: Social_Network_Ads.csv

User ID	Gender	Age	EstimatedSalary	Purchased
15624510	Male	19	19000	0
15810944	Male	35	20000	0
15668575	Female	26	43000	0
15603246	Female	27	57000	0
15804002	Male	19	76000	0
15728773	Male	27	58000	0
15598044	Female	27	84000	0
15694829	Female	32	150000	1
15600575	Male	25	33000	0
15727311	Female	35	65000	0

Output:



Compiled by: Ms. Beena Kapadia
Vidyalankar School of Information Technology

Practical no: 4b Naïve Bayes Classification

Code:

```
# Naive Bayes

# Importing the dataset
dataset = read.csv('D:\\2020\\Big Data Analytics\\Practical\\p4 naive
bayes\\Social_Network_Ads.csv')
dataset = dataset[3:5]

# Encoding the target feature as factor
dataset$Purchased = factor(dataset$Purchased, levels = c(0, 1))

# Splitting the dataset into the Training set and Test set
#install.packages('caTools')
library(caTools)
set.seed(123)
split = sample.split(dataset$Purchased, SplitRatio = 0.75)
training_set = subset(dataset, split == TRUE)
test_set = subset(dataset, split == FALSE)

# Feature Scaling
training_set[-3] = scale(training_set[-3])
test_set[-3] = scale(test_set[-3])

# Fitting Naive Bayes to the Training set
install.packages('e1071')
library(e1071)
classifier = naiveBayes(x = training_set[-3],
                        y = training_set$Purchased)

# Predicting the Test set results
y_pred = predict(classifier, newdata = test_set[-3])

# Making the Confusion Matrix
cm = table(test_set[, 3], y_pred)
print(cm)

# Visualising the Training set results
install.packages("ElemStatLearn")
library(ElemStatLearn)
set = training_set
print(set)
X1 = seq(min(set[, 1]) - 1, max(set[, 1]) + 1, by = 0.01)
X2 = seq(min(set[, 2]) - 1, max(set[, 2]) + 1, by = 0.01)
grid_set = expand.grid(X1, X2)
colnames(grid_set) = c('Age', 'EstimatedSalary')
y_grid = predict(classifier, newdata = grid_set)
plot(set[, -3],
    main = 'Naive Bayes (Training set)',
```

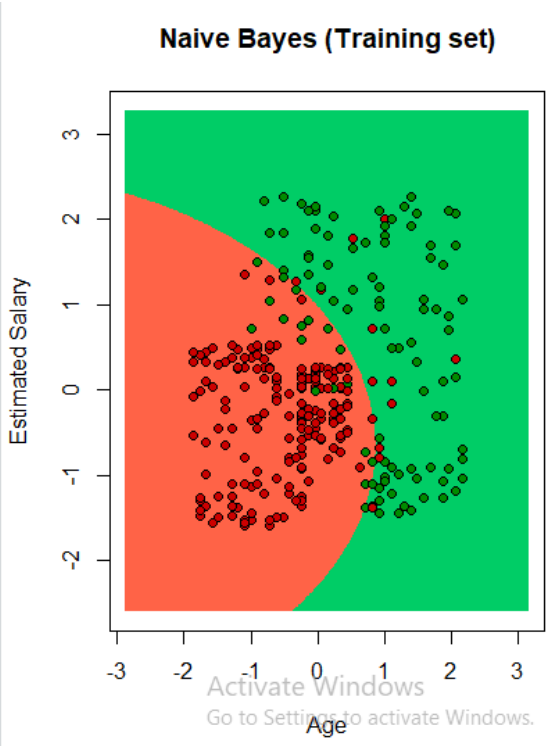
```
    xlab = 'Age', ylab = 'Estimated Salary',
    xlim = range(X1), ylim = range(X2))
contour(X1, X2, matrix(as.numeric(y_grid), length(X1), length(X2)), add = TRUE)
points(grid_set, pch = '.', col = ifelse(y_grid == 1, 'springgreen3', 'tomato'))
points(set, pch = 21, bg = ifelse(set[, 3] == 1, 'green4', 'red3'))

# Visualising the Test set results
library(ElemStatLearn)
set = test_set
X1 = seq(min(set[, 1]) - 1, max(set[, 1]) + 1, by = 0.01)
X2 = seq(min(set[, 2]) - 1, max(set[, 2]) + 1, by = 0.01)
grid_set = expand.grid(X1, X2)
colnames(grid_set) = c('Age', 'EstimatedSalary')
y_grid = predict(classifier, newdata = grid_set)
plot(set[, -3], main = 'NaiveBayes (Test set)',
    xlab = 'Age', ylab = 'Estimated Salary',
    xlim = range(X1), ylim = range(X2))
contour(X1, X2, matrix(as.numeric(y_grid), length(X1), length(X2)), add = TRUE)
points(grid_set, pch = '.', col = ifelse(y_grid == 1, 'springgreen3', 'tomato'))
points(set, pch = 21, bg = ifelse(set[, 3] == 1, 'green4', 'red3'))
```

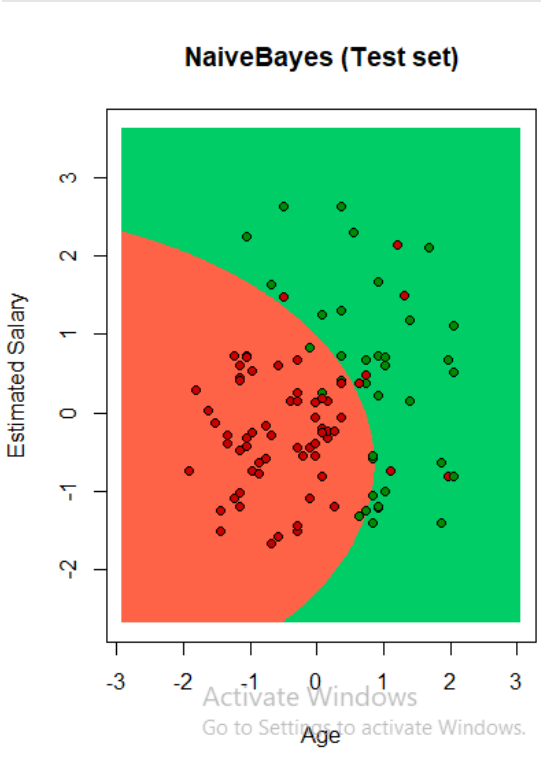
input: Social_Network_Ads.csv

User ID	Gender	Age	EstimatedSalary	Purchased
15624510	Male	19	19000	0
15810944	Male	35	20000	0
15668575	Female	26	43000	0
15603246	Female	27	57000	0
15804002	Male	19	76000	0
15728773	Male	27	58000	0
15598044	Female	27	84000	0
15694829	Female	32	150000	1
15600575	Male	25	33000	0
15727311	Female	35	65000	0

Output:

```
> classifier = naiveBayes(x = training_set[-3],
+                         y = training_set$Purchased)
> # Predicting the Test set results
> y_pred = predict(classifier, newdata = test_set[-3])
> # Making the Confusion Matrix
> cm = table(test_set[, 3], y_pred)
> # Making the Confusion Matrix
> cm = table(test_set[, 3], y_pred)
> print(cm)
   y_pred
     0  1
  0 57  7
  1  7 29
```





Practical 5: Text Analysis

Code:
Natural Language Processing

```
# Importing the dataset
dataset_original = read.delim('D:\\2020\\Big Data Analytics\\Practical\\P6
NLP\\Restaurant_Reviews.tsv', quote = '', stringsAsFactors = FALSE)

# Cleaning the texts
install.packages('tm')
install.packages('SnowballC')
library(tm)
library(SnowballC)
corpus = VCorpus(VectorSource(dataset_original$Review))
corpus = tm_map(corpus, content_transformer(tolower))
corpus = tm_map(corpus, removeNumbers)
corpus = tm_map(corpus, removePunctuation)
corpus = tm_map(corpus, removeWords, stopwords())
corpus = tm_map(corpus, stemDocument)
corpus = tm_map(corpus, stripWhitespace)


# Creating the Bag of Words model
dtm = DocumentTermMatrix(corpus)
dtm = removeSparseTerms(dtm, 0.999)
dataset = as.data.frame(as.matrix(dtm))
dataset$Liked = dataset_original$Liked
print(dataset$Liked)

# Encoding the target feature as factor
dataset$Liked = factor(dataset$Liked, levels = c(0, 1))

# Splitting the dataset into the Training set and Test set
install.packages('caTools')
library(caTools)
set.seed(123)
split = sample.split(dataset$Liked, SplitRatio = 0.8)
training_set = subset(dataset, split == TRUE)
test_set = subset(dataset, split == FALSE)

# Fitting Random Forest Classification to the Training set
install.packages('randomForest')
library(randomForest)
classifier = randomForest(x = training_set[-692],
                          y = training_set$Liked,
                          ntree = 10)

# Predicting the Test set results
y_pred = predict(classifier, newdata = test_set[-692])
```

```
# Making the Confusion Matrix
cm = table(test_set[, 692], y_pred)
print(cm)
```

input

Review Liked
Wow... Loved this place. 1
Crust is not good. 0
Not tasty and the texture was just nasty. 0
Stopped by during the late May bank holiday off Rick Steve recommendation and loved it. 1
The selection on the menu was great and so were the prices. 1
:
:
Overall I was not impressed and would not go back. 0
The whole experience was underwhelming, and I think we'll just go to Ninja Sushi next time. 0
Then, as if I hadn't wasted enough of my life there, they poured salt in the wound by drawing out the time it took to bring the check. 0

Output:

```
> print(cm)
   y_pred
     0  1
  0 82 18
  1 23 77
> |
```

Practical No.: 6 and 7
6 Aim: Install Virtual Box
7 Aim: Install, configure, and run Hadoop and HDFS ad explore HDFS.

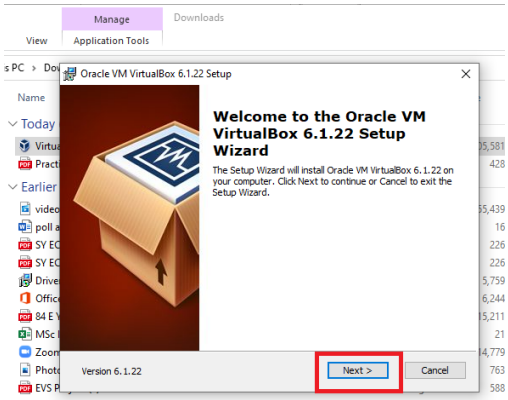
Practical No.: 1
Aim: Install, configure and run Hadoop and HDFS ad explore HDFS.

Step 1: Download and install VirtualBox
Go to the website of Oracle VirtualBox and get the latest stable version from the following site

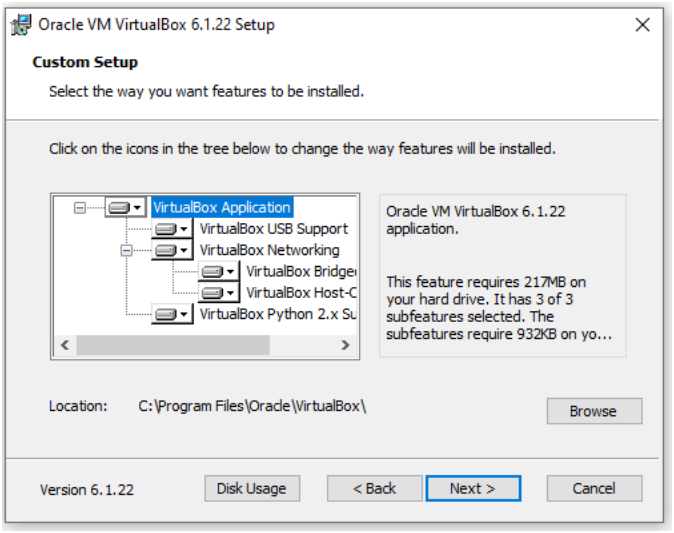
https://www.virtualbox.org/

click on 'Download''

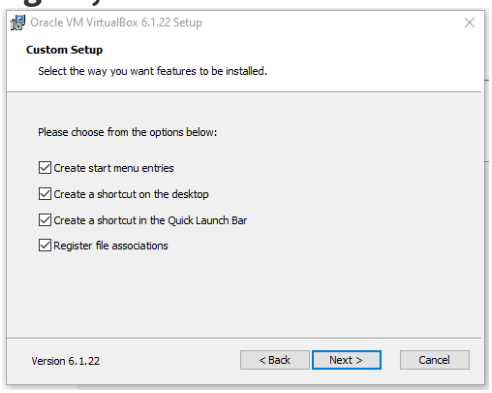
You will get VirtualBox-6.1.22-144080-Win.exe file downloaded. Double click and run it. Click on next.



Click on 'next' without changing the default folder as shown below:

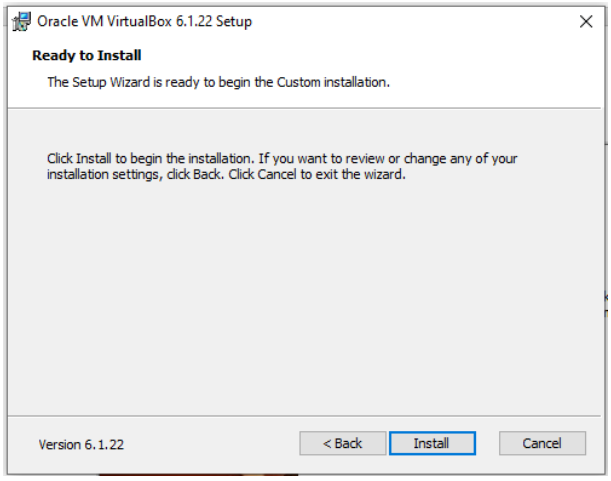


Again, click on next as shown below:



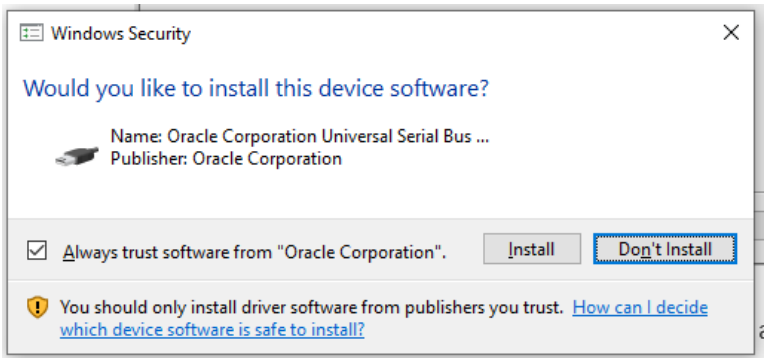
Finally, click on ‘Yes’.

Click on ’Install’.

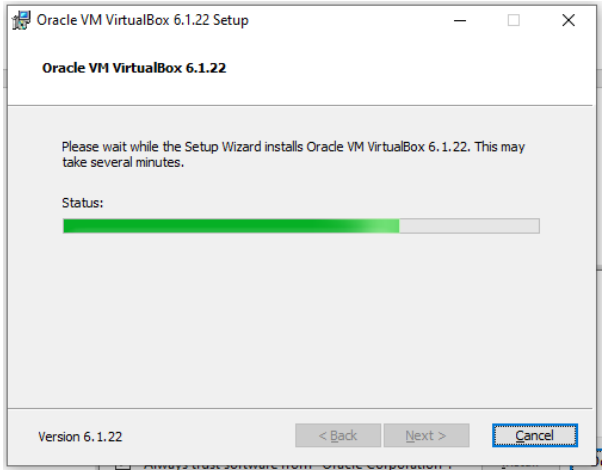


Compiled by: Ms. Beena Kapadia
Vidyalankar School of Information Technology

It may ask you for the permission to install, click 'yes' to allow.
Select 'Install' as shown below:

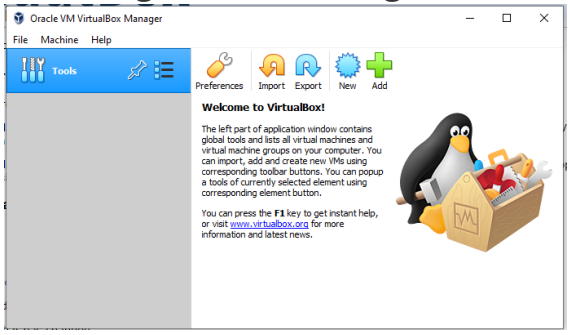


You will get the screen as shown below:



Click on 'Finish' to finish Installation of virtual box.

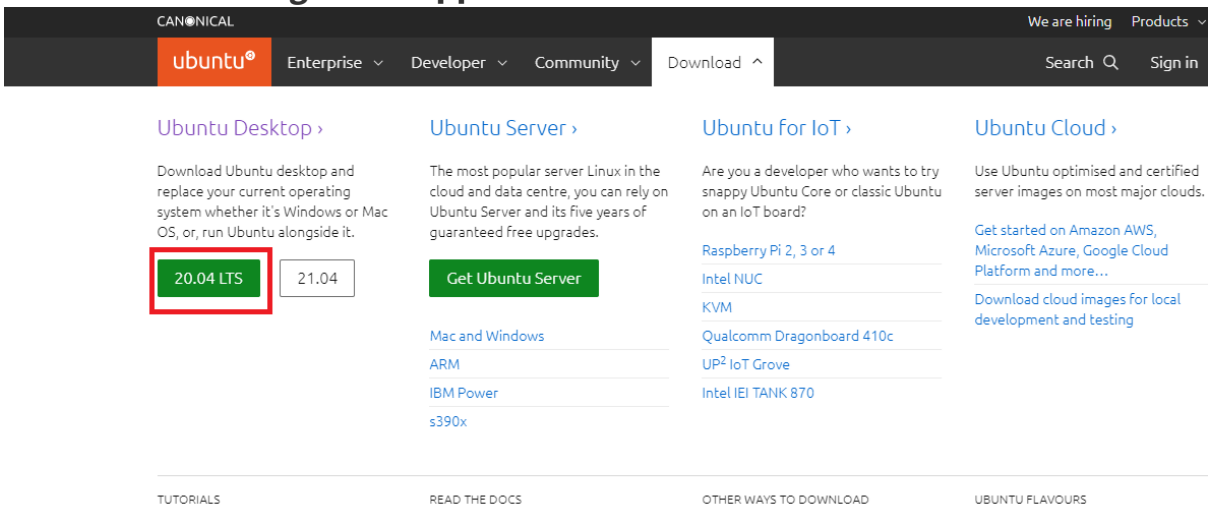
You will get the following screen:



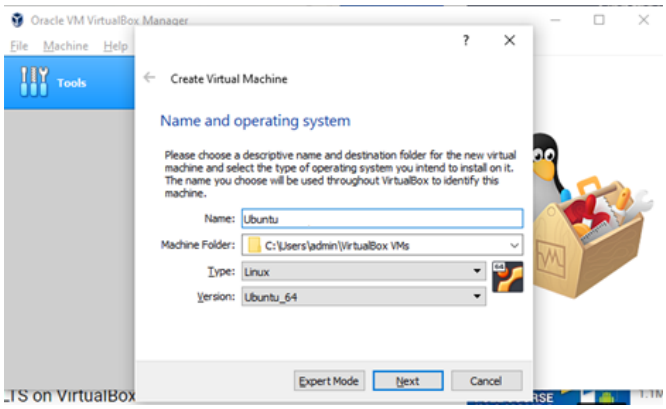
Compiled by: Ms. Beena Kapadia
Vidyalankar School of Information Technology

Step 2: download Ubuntu

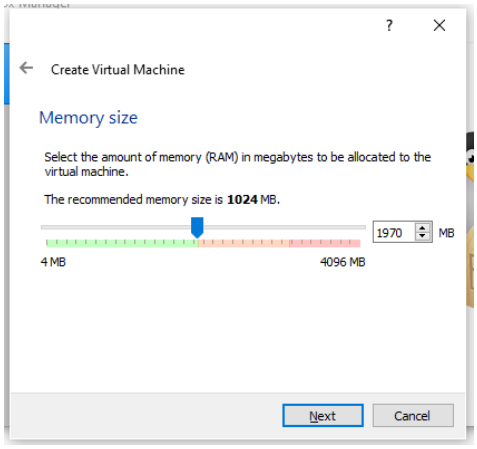
Download iso file ubuntu-20.04.2.0-desktop-amd64; which is required to install Ubuntu.
Browse ubuntu.com
Click on download and 20.04 LTS as shown below:
LTS stands for Long term support



You will get file, which may take few minutes to download.
Now, click on ‘New’ to virtual box and write Name as ‘Ubuntu’ as shown below:

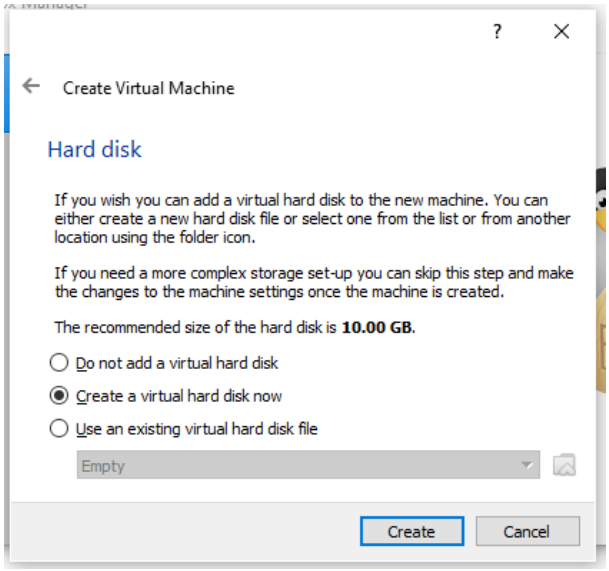


Click on ‘Next’.

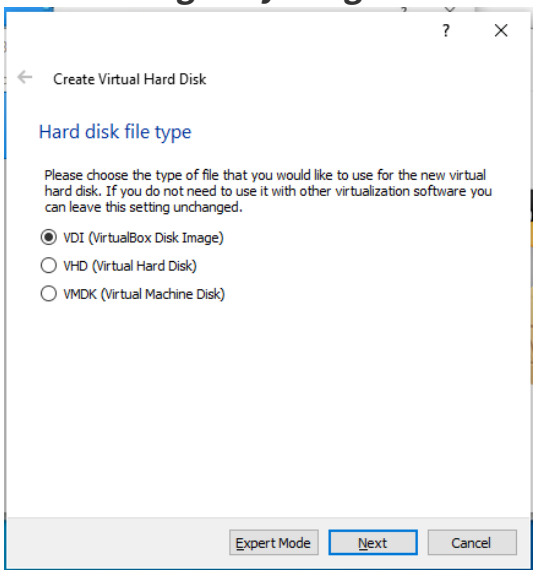


Compiled by: Ms. Beena Kapadia
Vidyalankar School of Information Technology

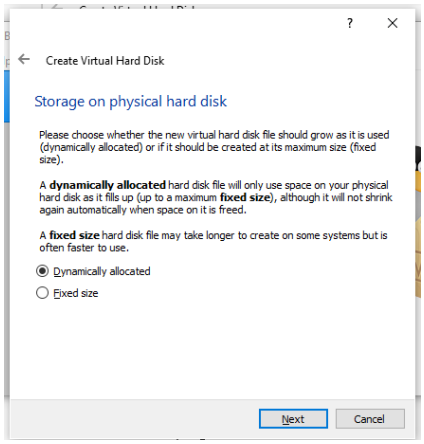
Here, you allow memory size up to green indicator (1970 MB). Click on 'Next'.



Don't change anything in this screen and click on 'Create'.

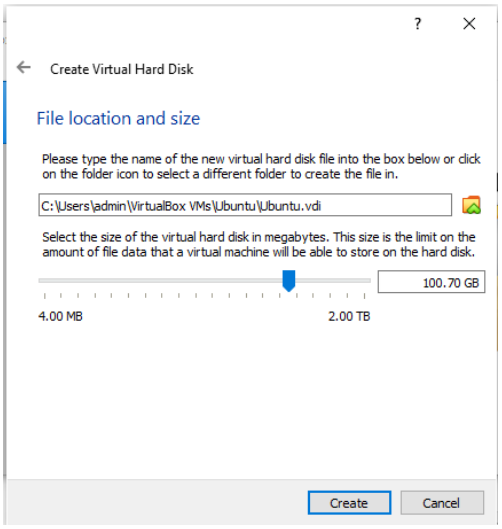


Click on 'Next', keeping the selection as it is (on VDI).'

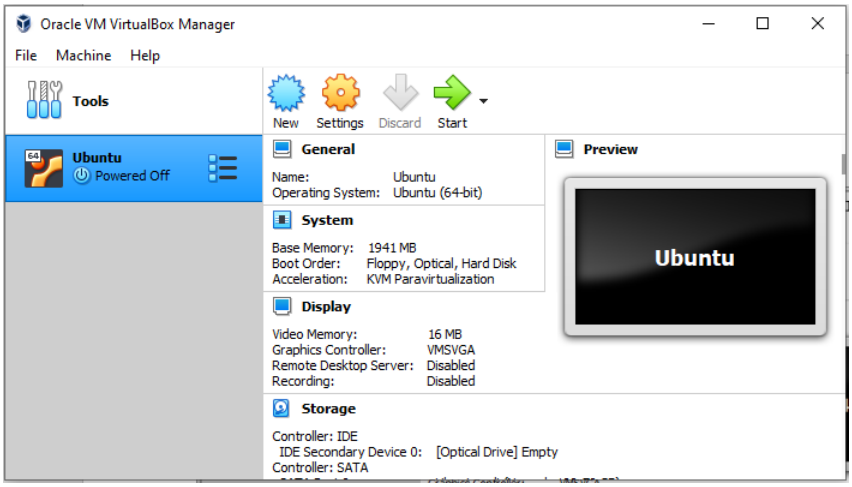


Compiled by: Ms. Beena Kapadia
Vidyalankar School of Information Technology

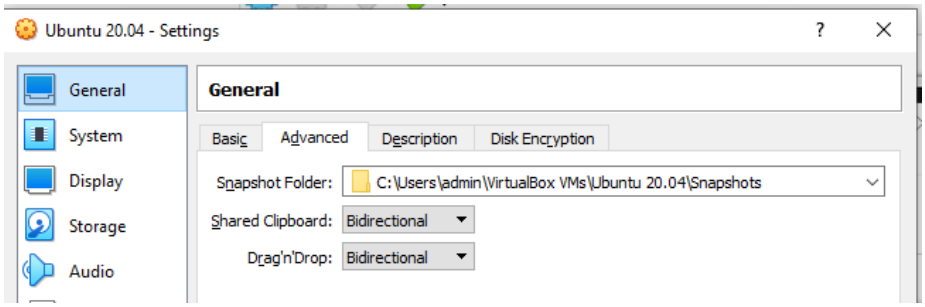
Keep this screen also as it is and click on ‘Next’.



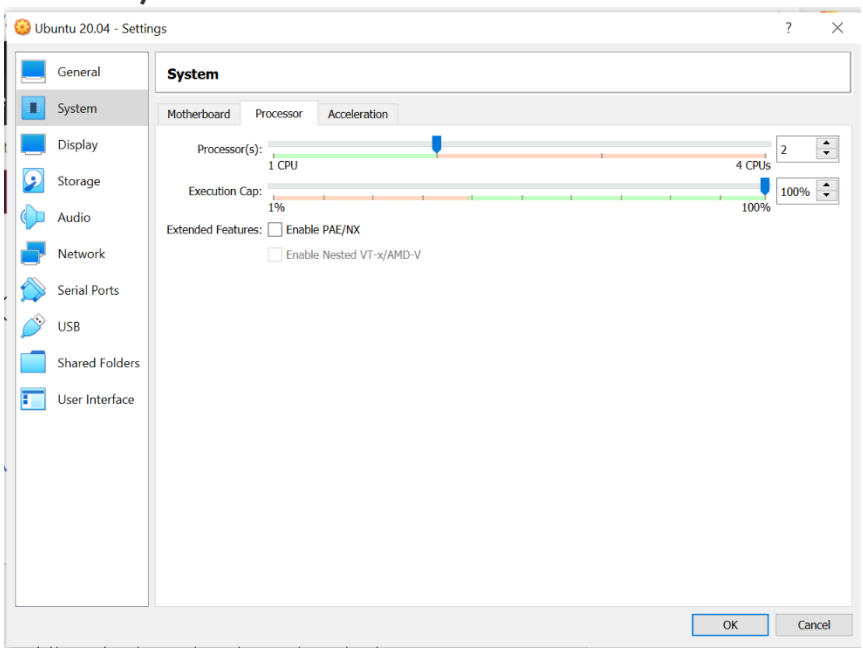
Keep the file location as it is but preferably keep size 100 GB and click on ‘Create’.
You may see the following screen having Ubuntu on Virtual Machine.



Select ‘settings’
Select ‘General’ -> ’ Basic’ as shown below:
You may change the name from Ubuntu to Ubuntu 20.04
Select bidirectional in ‘General’ -> ’ Advanced’ as shown below:



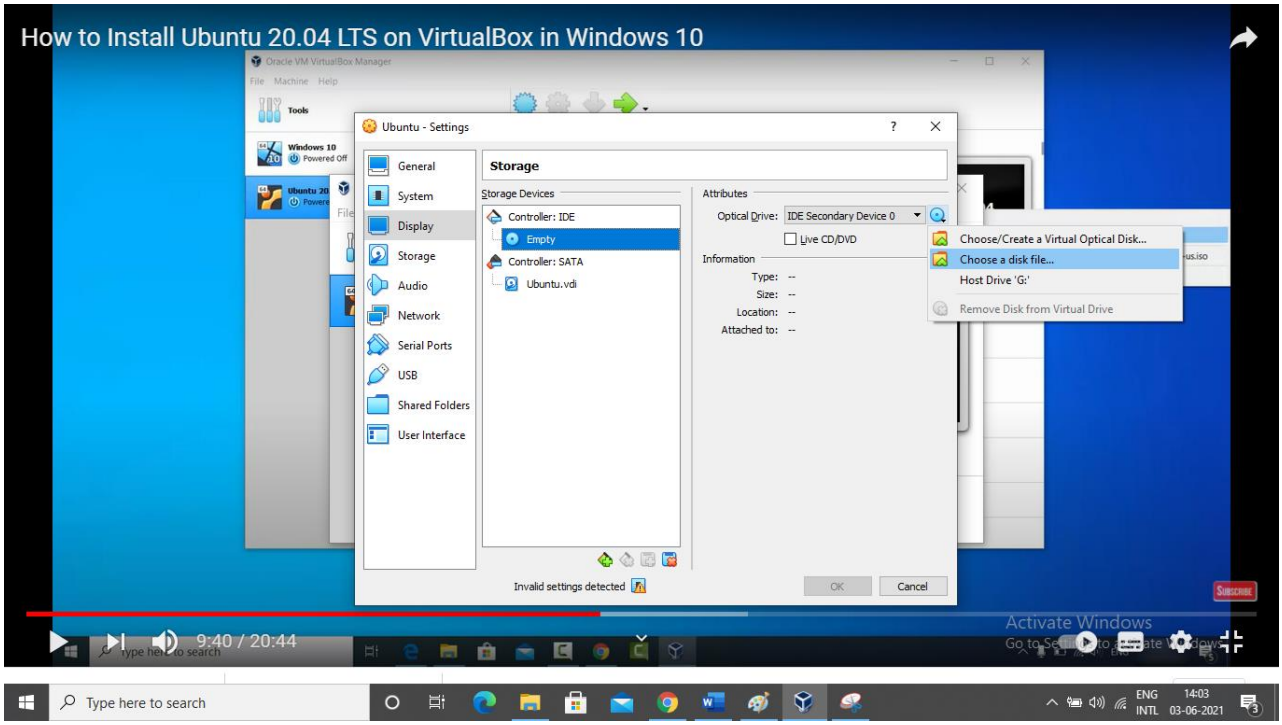
Go to ‘System’ option and change the processor up to green bar, usually 4.(if it allows)



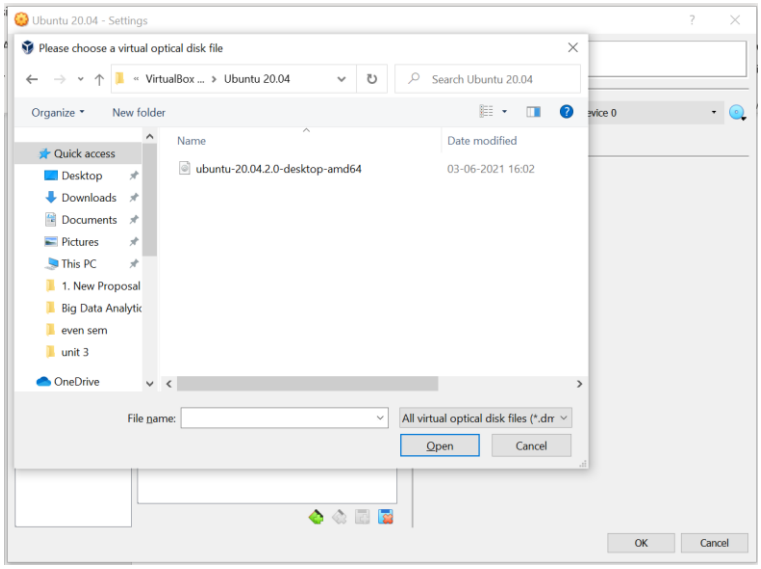
Cut and paste your ubuntu .iso file from current folder to C:\Users\ADMIN\VirtualBox VMs\Ubuntu 20.04 folder.

Click on ‘Storage’ and click on ‘Empty’ followed by ‘Choose a disk file’ as shown below:

Compiled by: Ms. Beena Kapadia
Vidyalankar School of Information Technology



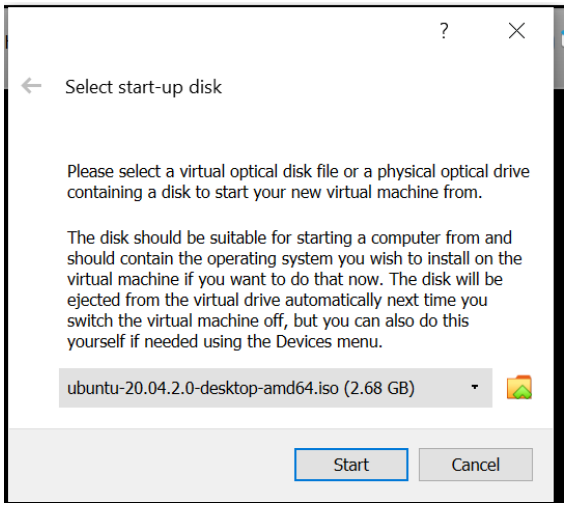
Browse the folder where you have selected ubuntu iso file.



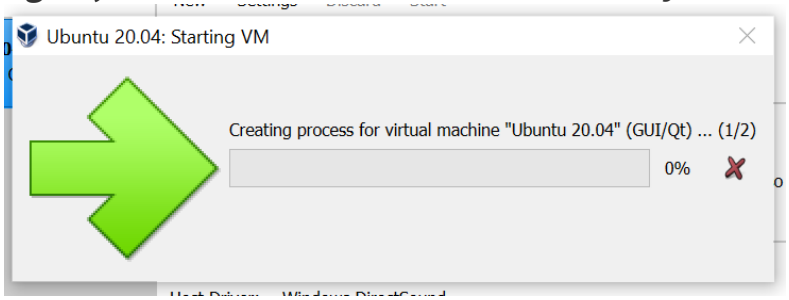
Click on Ubuntu….iso file and click on open and then click on ok.

Click on Ubuntu -> start button.

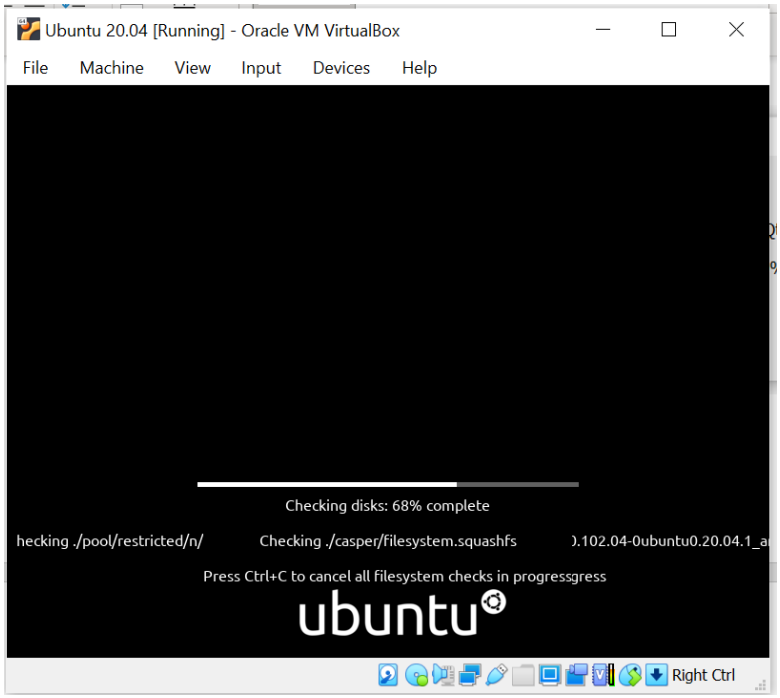
Compiled by: Ms. Beena Kapadia
Vidyalankar School of Information Technology



Again, click on 'Start' button. It will show you the following screen.

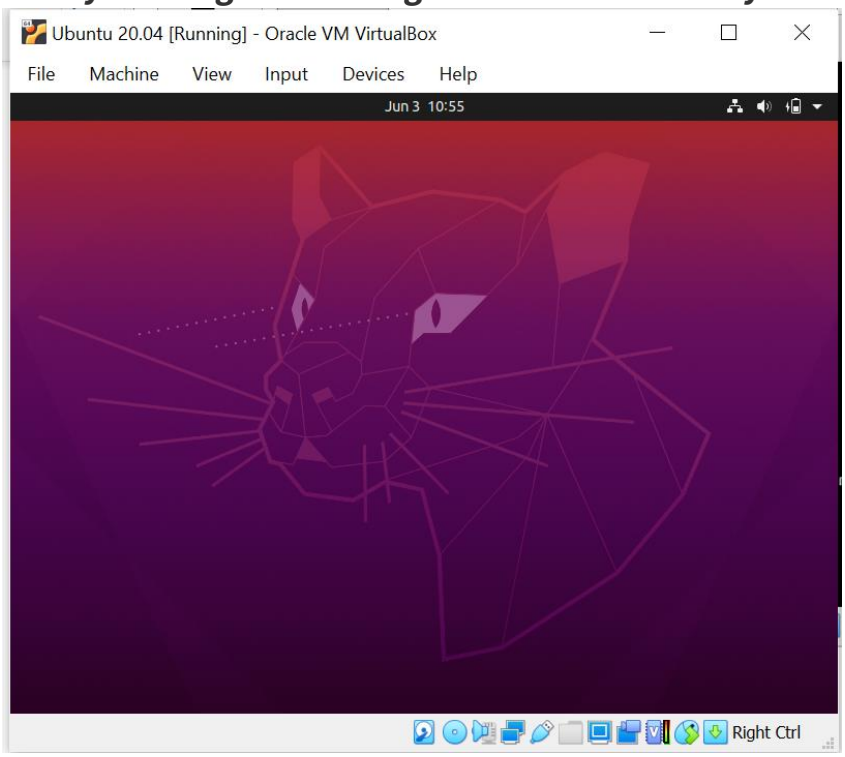


And simultaneously one more screen as follows:

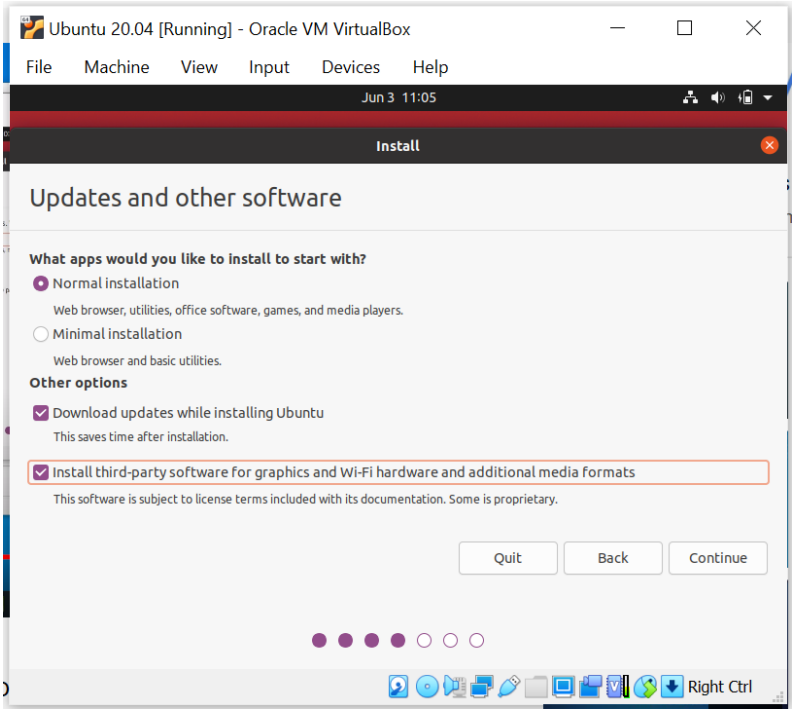


Compiled by: Ms. Beena Kapadia
Vidyalankar School of Information Technology

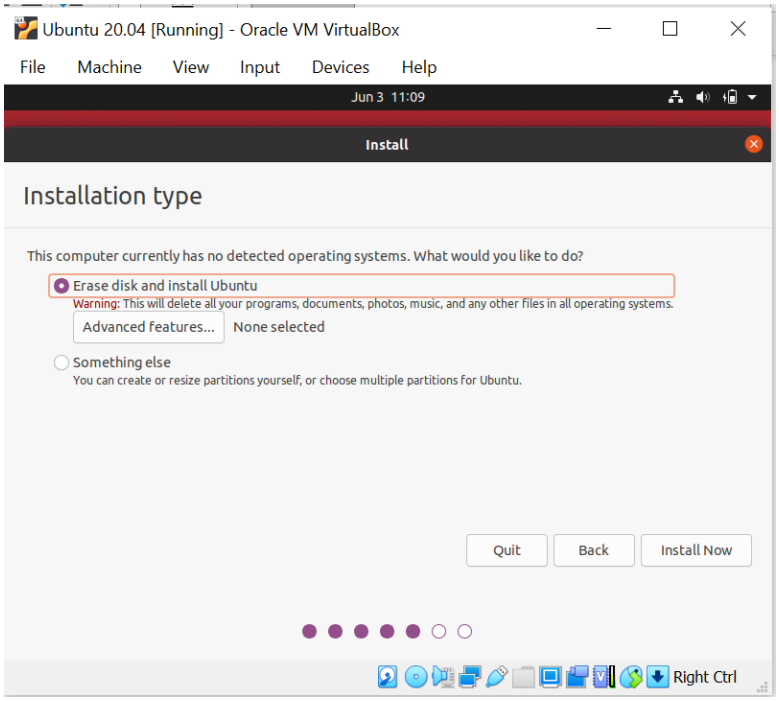
Keep on closing all warnings.
Next you will get following screen automatically.



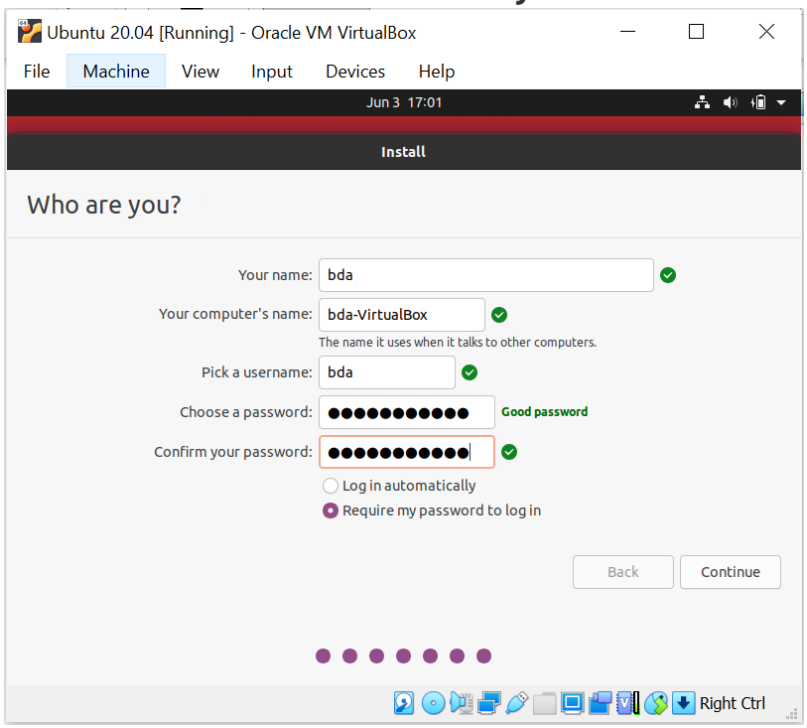
Select language -> English and click on 'Install Ubuntu'.in 'Keyboard Layout' screen, select 'English UK'. Click on 'Continue'.
Select the checkbox for third party software as shown below:



Click on 'continue'.

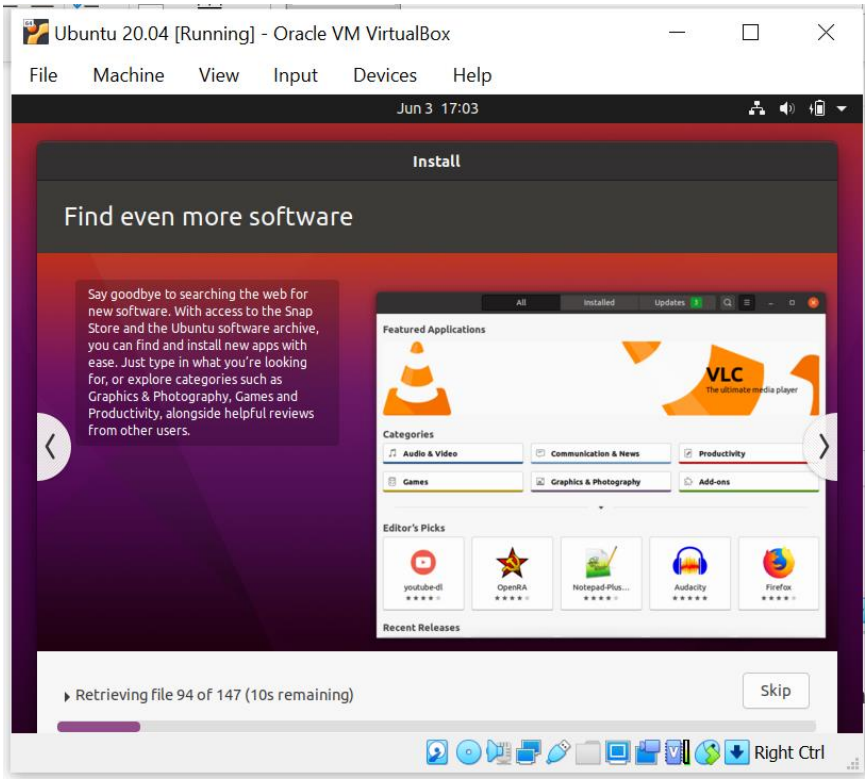


Select Erase disk and Install Ubuntu and click on 'Install Now'.
Click on 'Continue' on the next screen.
Select "Kolkata" for "where are you?" and click on 'Continue'.



Click on continue after entering name, company name, username, password and confirm your password.

Compiled by: Ms. Beena Kapadia
Vidyalankar School of Information Technology



Installation of Ubuntu started. Click on finish once installation done. Click on restart and press Enter key.

Step 3 Install Hadoop

Login to ubuntu
Some keys may change like you try to type @ and it types “.
** please refer to note - Some Keys for Ubuntu under UK keyboard layout – at the end.

Search for Ubuntu terminal on search bar, after login done.

Apply following commands from ubuntu terminal

Prerequisite

buntu@ubuntu:~$ sudo apt update
Ign:1 cdrom://Ubuntu 20.04.2.0 LTS _Focal Fossa_ - Release amd64 (20210209.1) focal InRelease
Hit:2 cdrom://Ubuntu 20.04.2.0 LTS _Focal Fossa_ - Release amd64 (20210209.1) focal Release
Hit:4 http://archive.ubuntu.com/ubuntu focal InRelease
Hit:5 http://archive.ubuntu.com/ubuntu focal-updates InRelease
Hit:6 http://security.ubuntu.com/ubuntu focal-security InRelease
Reading package lists... Done

Building dependency tree
Reading state information... Done
291 packages can be upgraded. Run 'apt list --upgradable' to see them.

bda@bda-VirtualBox:~$ sudo apt install default-jdk
Reading package lists... Done
Building dependency tree
:
etting up default-jdk (2:1.11-72) ...
Setting up libxt-dev:amd64 (1:1.1.5-1) ...

bda@bda-VirtualBox:~$ java -version
openjdk version "11.0.11" 2021-04-20
OpenJDK Runtime Environment (build 11.0.11+9-Ubuntu-0ubuntu2.20.04)
OpenJDK 64-Bit Server VM (build 11.0.11+9-Ubuntu-0ubuntu2.20.04, mixed mode, sharing)

open ssh server
bda@bda-VirtualBox:~$ sudo apt install openssh-server openssh-client -y
Reading package lists... Done
Building dependency tree
:
Processing triggers for ufw (0.36-6) ...

bda@bda-VirtualBox:~$ sudo adduser hdoop
Adding user `hdoop' ...
Adding new group `hdoop' (1000) ...
Adding new user `hdoop' (1000) with group `hdoop' ...
Creating home directory `/home/hdoop' ...
Copying files from `/etc/skel' ...
New password: hdoop
Retype new password:
passwd: password updated successfully
Changing the user information for hdoop
Enter the new value, or press ENTER for the default
    Full Name []:
    Room Number []:
    Work Phone []:
    Home Phone []:
    Other []:
Is the information correct? [Y/n] y

bda@bda-VirtualBox:~$ su - hdoop
Password: hdoop

hdoop@bda-VirtualBox:~$ ssh-keygen -t rsa -P '' -f ~/.ssh/id_rsa
Generating public/private rsa key pair.
Created directory '/home/hdoop/.ssh'.
Your identification has been saved in /home/hdoop/.ssh/id_rsa
Your public key has been saved in /home/hdoop/.ssh/id_rsa.pub
The key fingerprint is:

SHA256:EDxiHTL1r3LUCdKFWc0moPHUh1D8tU6Y0b2rnxuwUtQ hdoop@bda-VirtualBox
The key's randomart image is:
+---[RSA 3072]----+
| o+=.+X++ . . |
| oo+Oo.= * + .|
| . .+.= = * E.|
| o + .= o. |
| S + = .|
| . . . +. |
| . o . ... |
| o   .. o|
| .+.|
+----[SHA256]-----+

hdoop@bda-VirtualBox:~$ cat ~/.ssh/id_rsa.pub >> ~/.ssh/authorized_keys
hdoop@bda-VirtualBox:~$ chmod 0600 ~/.ssh/authorized_keys

hdoop@bda-VirtualBox:~$ ssh localhost
The authenticity of host 'localhost (127.0.0.1)' can't be established.
ECDSA key fingerprint is
SHA256:4TE4DDAv14vhARPWjZcW3C5UM3X94B7wUudPrT+ZmF0.
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
:
Ubuntu comes with ABSOLUTELY NO WARRANTY, to the extent permitted by applicable law.

Downloading Hadoop

hdoop@bda-VirtualBox:~$ wget
https://downloads.apache.org/hadoop/common/hadoop-3.3.1/hadoop-3.3.1.tar.gz
--2021-06-14 08:52:00-- https://downloads.apache.org/hadoop/common/hadoop-3.3.1/hadoop-3.3.1.tar.gz
Resolving downloads.apache.org (downloads.apache.org)... 88.99.95.219, 135.181.209.10, 135.181.214.104, ...
Connecting to downloads.apache.org (downloads.apache.org)|88.99.95.219|:443... connected.
HTTP request sent, awaiting response... 200 OK
Length: 359196911 (343M) [application/x-gzip]
Saving to: ‘hadoop-3.3.1.tar.gz’

hadoop-3.3.1.tar.gz 100%[==================>] 342.56M 15.4MB/s in 33s

2021-06-14 08:52:34 (10.2 MB/s) - ‘hadoop-3.3.1.tar.gz’ saved [359196911/359196911]

hdoop@bda-VirtualBox:~$ ls
hadoop-3.3.1.tar.gz
hdoop@bda-VirtualBox:~$ tar xzf hadoop-3.3.1.tar.gz
hdoop@bda-VirtualBox:~$ ls
hadoop-3.3.1 hadoop-3.3.1.tar.gz

Editing 6 important files for creating a single cluster

hdoop@bda-VirtualBox:~$ su - bda
bda@bda-VirtualBox:~$ sudo adduser hdoop sudo
Adding user `hdoop' to group `sudo' ...
Adding user hdoop to group sudo
Done.
bda@bda-VirtualBox:~$ su - hdoop

1.
hdoop@bda-VirtualBox:~$ sudo nano .bashrc
File will be opened and add following lines at the end of the file:

```
#Hadoop Related Options
export HADOOP_HOME=/home/hdoop/hadoop-3.3.1
export HADOOP_INSTALL=$HADOOP_HOME
export HADOOP_MAPRED_HOME=$HADOOP_HOME
export HADOOP_COMMON_HOME=$HADOOP_HOME
export HADOOP_HDFS_HOME=$HADOOP_HOME
export YARN_HOME=$HADOOP_HOME
export HADOOP_COMMON_LIB_NATIVE_DIR=$HADOOP_HOME/lib/native
export PATH=$PATH:$HADOOP_HOME/sbin:$HADOOP_HOME/bin
export HADOOP_OPTS="-Djava.library.path=$HADOOP_HOME/lib/nativ"
```

save this file as ctrl x and y. Press enter.

hdoop@bda-VirtualBox:~$ source ~/.bashrc

2.

Edit hadoop-env.sh File

The *hadoop-env.sh* file serves as a master file to configure YARN, HDFS, MapReduce, and Hadoop-related project settings.

When setting up a **single node Hadoop cluster**, you need to define which Java implementation is to be utilized. Use the previously created **$HADOOP_HOME** variable to access the *hadoop-env.sh* file:

hdoop@bda-VirtualBox:~$ sudo nano $HADOOP_HOME/etc/hadoop/hadoop-env.sh
at the end of the file add the following line
export JAVA_HOME=/usr/lib/jvm/java-11-openjdk-amd64/
save it.

Edit core-site.xml File

The *core-site.xml* file defines HDFS and Hadoop core properties.

To set up Hadoop in a pseudo-distributed mode, you need to **specify the URL** for your NameNode, and the temporary directory Hadoop uses for the map and reduce process.

Open the *core-site.xml* file in a text editor:

hdoop@bda-VirtualBox:~$ sudo nano $HADOOP_HOME/etc/hadoop/core-site.xml

```
<configuration>
<property>
 <name>hadoop.tmp.dir</name>
 <value>/home/hdoop/tmpdata</value>
</property>
<property>
 <name>fs.default.name</name>
 <value>hdfs://localhost:9000</value>
</property>
</configuration>
```

4

hdoop@bda-VirtualBox:~$ sudo nano $HADOOP_HOME/etc/hadoop/hdfs-site.xml

```
<configuration>
<property>
 <name>dfs.data.dir</name>
 <value>/home/hdoop/dfsdata/namenode</value>
</property>
<property>
 <name>dfs.data.dir</name>
 <value>/home/hdoop/dfsdata/datanode</value>
</property>
<property>
 <name>dfs.replication</name>
 <value>1</value>
</property>
</configuration>
```

5

hdoop@bda-VirtualBox:~$ sudo nano $HADOOP_HOME/etc/hadoop/mapred-site.xml

```
<configuration>
<property>
 <name>mapreduce.framework.name</name>
 <value>yarn</value>
</property>
</configuration>
```

6

hdoop@bda-VirtualBox:~$ sudo nano $HADOOP_HOME/etc/hadoop/yarn-site.xml

```
<configuration>
<property>
 <name>yarn.nodemanager.aux-services</name>
 <value>mapreduce_shuffle</value>
</property>
<property>
 <name>yarn.nodemanager.aux-services.mapreduce.shuffle.class</name>
 <value>org.apache.hadoop.mapred.ShuffleHandler</value>
</property>
<property>
 <name>yarn.resourcemanager.hostname</name>
 <value>127.0.0.1</value>
</property>
```

```
<property>
  <name>yarn.acl.enable</name>
  <value>0</value>
</property>
<property>
  <name>yarn.nodemanager.env-whitelist</name>

<value>JAVA_HOME,HADOOP_COMMON_HOME,HADOOP_HDFS_HOME,HADOOP
_CONF_DIR,CLASSPATH_PERPEND_DISTCACHE,HADOOP_YARN_HOME,HADOO
P_MAPRED_HOME</value>
</property>
</configuration>
```

Format HDFS NameNode

hdoop@bda-VirtualBox:~$ hdfs namenode -format

```
:
xid=0 when meet shutdown.
2021-06-18 14:16:33,353 INFO namenode.NameNode: SHUTDOWN_MSG:
/************************************************************
SHUTDOWN_MSG: Shutting down NameNode at bda-VirtualBox/127.0.1.1
************************************************************/
```

Start Hadoop Cluster (services)

hdoop@bda-VirtualBox:~$ cd Hadoop-3.3.1
hdoop@bda-VirtualBox:~/Hadoop-3.3.1$ cd sbin
hdoop@bda-VirtualBox:~/hadoop-3.3.1/sbin$ ./start-dfs.sh

```
Starting namenodes on [localhost]
Starting datanodes
Starting secondary namenodes [bda-VirtualBox]
bda-VirtualBox: Warning: Permanently added 'bda-virtualbox' (ECDSA) to the list of known
hosts.
2021-06-18 14:26:34,962 WARN util.NativeCodeLoader: Unable to load native-hadoop
library for your platform... using builtin-java classes where applicable
```

```
hdoop@bda-VirtualBox:~/hadoop-3.3.1/sbin$ ./start-yarn.sh
Starting resourcemanager
Starting nodemanagers
To see all components, we use jps command:
```

hdoop@bda-VirtualBox:~/hadoop-3.3.1/sbin$ jps

```
11744 NodeManager
11616 ResourceManager
12192 Jps
11268 SecondaryNameNode
11077 DataNode
10954 NameNode
```

hdoop@bda-VirtualBox:~/hadoop-3.3.1/sbin$ hdfs dfs -ls /
2021-06-18 14:33:24,698 WARN util.NativeCodeLoader: Unable to load native-hadoop library for your platform... using builtin-java classes where applicable

hdoop@bda-VirtualBox:~/hadoop-3.3.1/sbin$ sudo nano /home/bda/sample.txt
[sudo] password for hdoop:
edit the file by adding some text and save and exit

hdoop@bda-VirtualBox:~/hadoop-3.3.1/sbin$ ls /home/bda/
Desktop Downloads Pictures sample.txt Videos
Documents Music Public Templates

hdoop@bda-VirtualBox:~/hadoop-3.3.1/sbin$ hdfs dfs -put /home/bda/sample.txt /
2021-06-18 14:44:24,257 WARN util.NativeCodeLoader: Unable to load native-hadoop library for your platform... using builtin-java classes where applicable

doop@bda-VirtualBox:~/hadoop-3.3.1/sbin$ hdfs dfs -ls /
2021-06-18 14:48:17,221 WARN util.NativeCodeLoader: Unable to load native-hadoop library for your platform... using builtin-java classes where applicable
Found 1 items
-rw-r--r-- 1 hdoop supergroup 6 2021-06-18 14:44 /sample.txt

****Note:**
Some Keys for Ubuntu under UK keyboard layout
“ -> @
@ -> “
pipe -> take from this file or on google search for pipe in linux
~ -> pipe

Compiled by: Ms. Beena Kapadia
Vidyalankar School of Information Technology