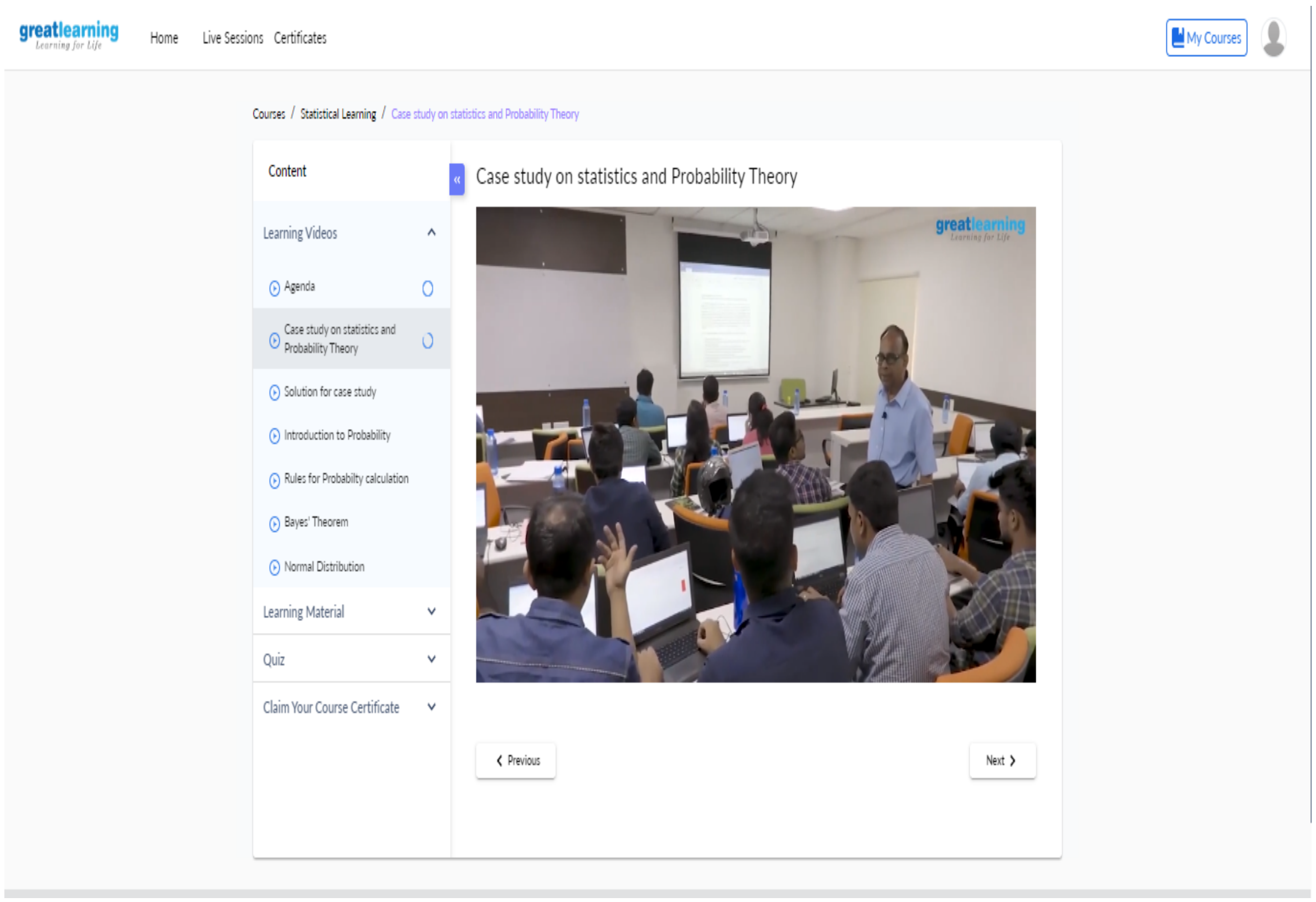
DAILY ASSESSMENT FORMAT

Date:	16/06/2020	Name:	Akshatha M Deshpande
Course:	Great Learning	USN:	4AL17EC006
Topic:	Case Study on statistics & probability theory,solutions for case study	Semester & Section:	6th Sem A sec
Github Repository:	AkshathaDeshpande		

FORENOON SESSION DETAILS

Image of session


greatlearning Learning for Life
Home
Live Sessions
Certificates
My Courses
Courses / Statistical Learning / Case study on statistics and Probability Theory
Content
Learning Videos
Agenda
Case study on statistics and Probability Theory
Solution for case study
Introduction to Probability
Rules for Probabilty calculation
Bayes' Theorem
Normal Distribution
Learning Material
Quiz
Claim Your Course Certificate
Case study on statistics and Probability Theory
Previous
Next

Report – Report can be typed or hand written for up to two pages.

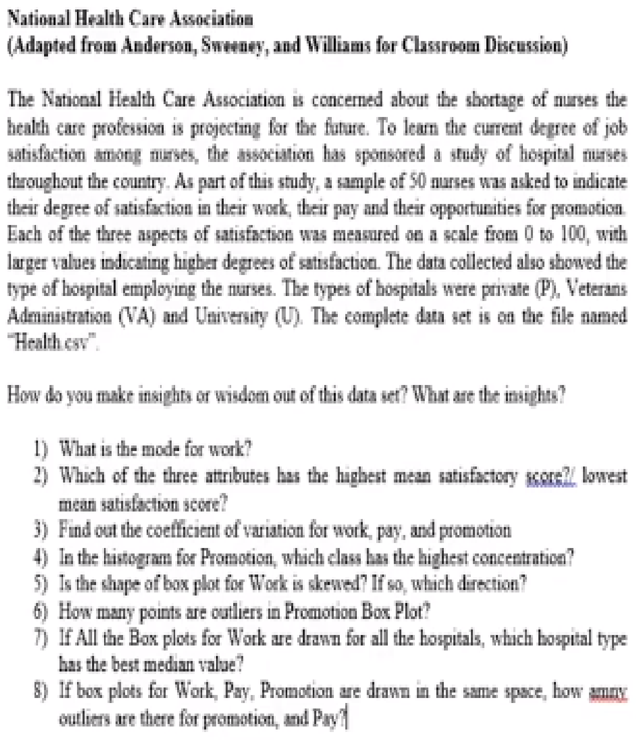
AGENDA:

- Case study for statistics
- Probability and its types
- Bayes theorem
- Normal distribution and bell curve

Steps:

- Identify and define the research questions
- Select the cases
- Collect data
- Evaluate and analyze the data
- Result presentation

Questions:



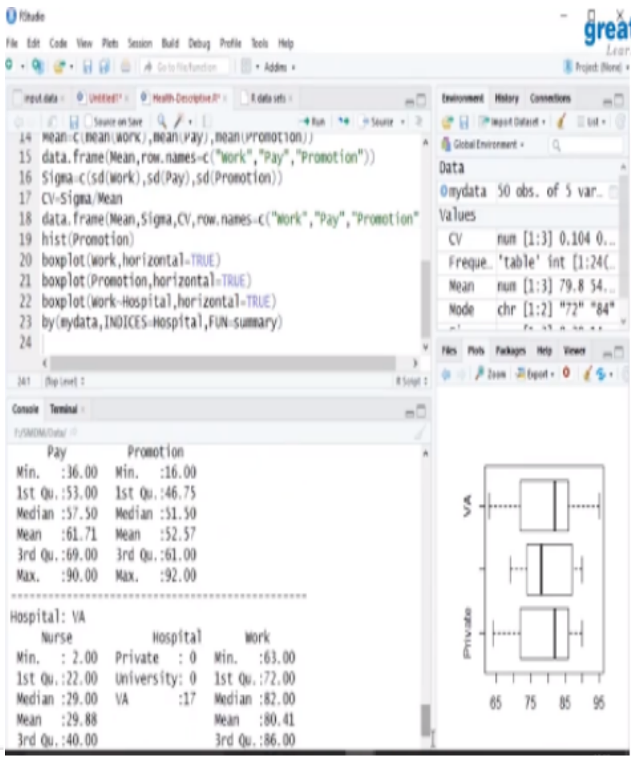
National Health Care Association
(Adapted from Anderson, Sweeney, and Williams for Classroom Discussion)

The National Health Care Association is concerned about the shortage of nurses the health care profession is projecting for the future. To learn the current degree of job satisfaction among nurses, the association has sponsored a study of hospital nurses throughout the country. As part of this study, a sample of 50 nurses was asked to indicate their degree of satisfaction in their work, their pay and their opportunities for promotion. Each of the three aspects of satisfaction was measured on a scale from 0 to 100, with larger values indicating higher degrees of satisfaction. The data collected also showed the type of hospital employing the nurses. The types of hospitals were private (P), Veterans Administration (VA) and University (U). The complete data set is on the file named "Health.csv".

How do you make insights or wisdom out of this data set? What are the insights?

1) What is the mode for work?
2) Which of the three attributes has the highest mean satisfactory score?/ lowest mean satisfaction score?
3) Find out the coefficient of variation for work, pay, and promotion
4) In the histogram for Promotion, which class has the highest concentration?
5) Is the shape of box plot for Work is skewed? If so, which direction?
6) How many points are outliers in Promotion Box Plot?
7) If All the Box plots for Work are drawn for all the hospitals, which hospital type has the best median value?
8) If box plots for Work, Pay, Promotion are drawn in the same space, how amny outliers are there for promotion, and Pay?

Solutions of case study:



```
Mean=c(mean(Work),mean(Pay),mean(Promotion))
data.frame(Mean,row.names=c("Work","Pay","Promotion"))
Sigma=c(sd(Work),sd(Pay),sd(Promotion))
CV=Sigma/Mean
data.frame(Mean,Sigma,CV,row.names=c("Work","Pay","Promotion"
hist(Promotion)
boxplot(Work,horizontal=TRUE)
boxplot(Promotion,horizontal=TRUE)
boxplot(Work~Hospital,horizontal=TRUE)
by(mydata,INDICES=Hospital,FUN=summary)
```

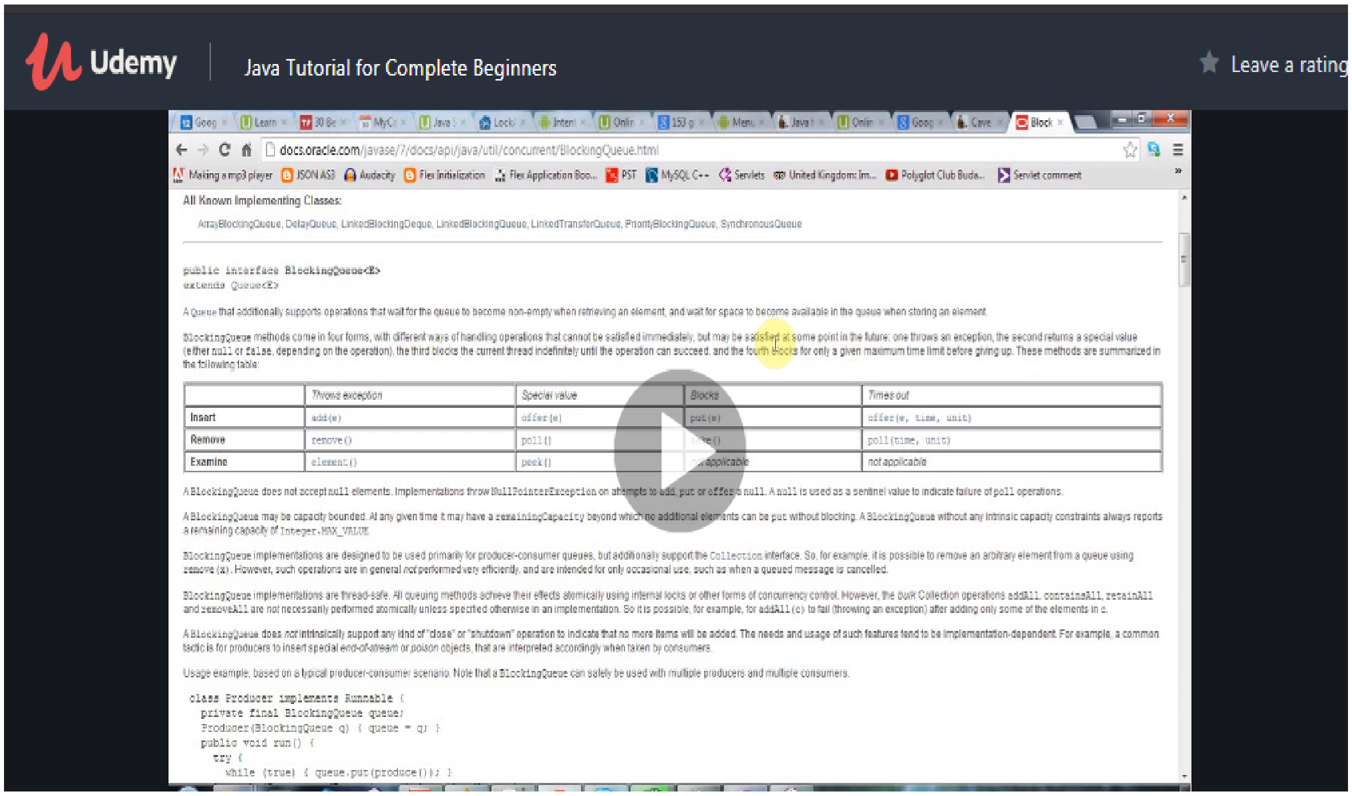
```
     Pay            Promotion
 Min.   :36.00   Min.   :16.00
 1st Qu.:53.00   1st Qu.:46.75
 Median :57.50   Median :51.50
 Mean   :61.71   Mean   :52.57
 3rd Qu.:69.00   3rd Qu.:61.00
 Max.   :90.00   Max.   :92.00
------------------------------------------------
Hospital: VA
     Nurse              Hospital       Work
 Min.   : 2.00   Private   : 0   Min.   :63.00
 1st Qu.:22.00   University: 0   1st Qu.:72.00
 Median :29.00   VA        :17   Median :82.00
 Mean   :29.88                   Mean   :80.41
 3rd Qu.:40.00                   3rd Qu.:86.00
```

```
Data
mydata   50 obs. of 5 var..
Values
CV       num [1:3] 0.104 0...
Freque.. 'table' int [1:24(..
Mean     num [1:3] 79.8 54...
Mode     chr [1:2] "72" "84"
```

Date:	16/06/2020	Name:	Akshatha M Deshpande
Course:	Java	USN:	4AL17EC006
Topic:	Programming	Semester & Section:	6th Sem A sec

AFTERNOON SESSION DETAILS

Image of session


Udemy
Java Tutorial for Complete Beginners
Leave a rating
docs.oracle.com/javase/7/docs/api/java/util/concurrent/BlockingQueue.html

All Known Implementing Classes:

ArrayBlockingQueue, DelayQueue, LinkedBlockingDeque, LinkedBlockingQueue, LinkedTransferQueue, PriorityBlockingQueue, SynchronousQueue

```
public interface BlockingQueue<E>
extends Queue<E>
```

A Queue that additionally supports operations that wait for the queue to become non-empty when retrieving an element, and wait for space to become available in the queue when storing an element.

BlockingQueue methods come in four forms, with different ways of handling operations that cannot be satisfied immediately, but may be satisfied at some point in the future: one throws an exception, the second returns a special value (either null or false, depending on the operation), the third blocks the current thread indefinitely until the operation can succeed, and the fourth blocks for only a given maximum time limit before giving up. These methods are summarized in the following table:

	Throws exception	*Special value*	*Blocks*	*Times out*
Insert	add(e)	offer(e)	put(e)	offer(e, time, unit)
Remove	remove()	poll()	take()	poll(time, unit)
Examine	element()	peek()	not applicable	not applicable

A BlockingQueue does not accept null elements. Implementations throw NullPointerException on attempts to add, put or offer a null. A null is used as a sentinel value to indicate failure of poll operations.

A BlockingQueue may be capacity bounded. At any given time it may have a remainingCapacity beyond which no additional elements can be put without blocking. A BlockingQueue without any intrinsic capacity constraints always reports a remaining capacity of Integer.MAX_VALUE.

BlockingQueue implementations are designed to be used primarily for producer-consumer queues, but additionally support the Collection interface. So, for example, it is possible to remove an arbitrary element from a queue using remove(x). However, such operations are in general *not* performed very efficiently, and are intended for only occasional use, such as when a queued message is cancelled.

BlockingQueue implementations are thread-safe. All queuing methods achieve their effects atomically using internal locks or other forms of concurrency control. However, the *bulk* Collection operations addAll, containsAll, retainAll and removeAll are not necessarily performed atomically unless specified otherwise in an implementation. So it is possible, for example, for addAll(c) to fail (throwing an exception) after adding only some of the elements in c.

A BlockingQueue does *not* intrinsically support any kind of "close" or "shutdown" operation to indicate that no more items will be added. The needs and usage of such features tend to be implementation-dependent. For example, a common tactic is for producers to insert special *end-of-stream* or *poison* objects, that are interpreted accordingly when taken by consumers.

Usage example, based on a typical producer-consumer scenario. Note that a BlockingQueue can safely be used with multiple producers and multiple consumers.

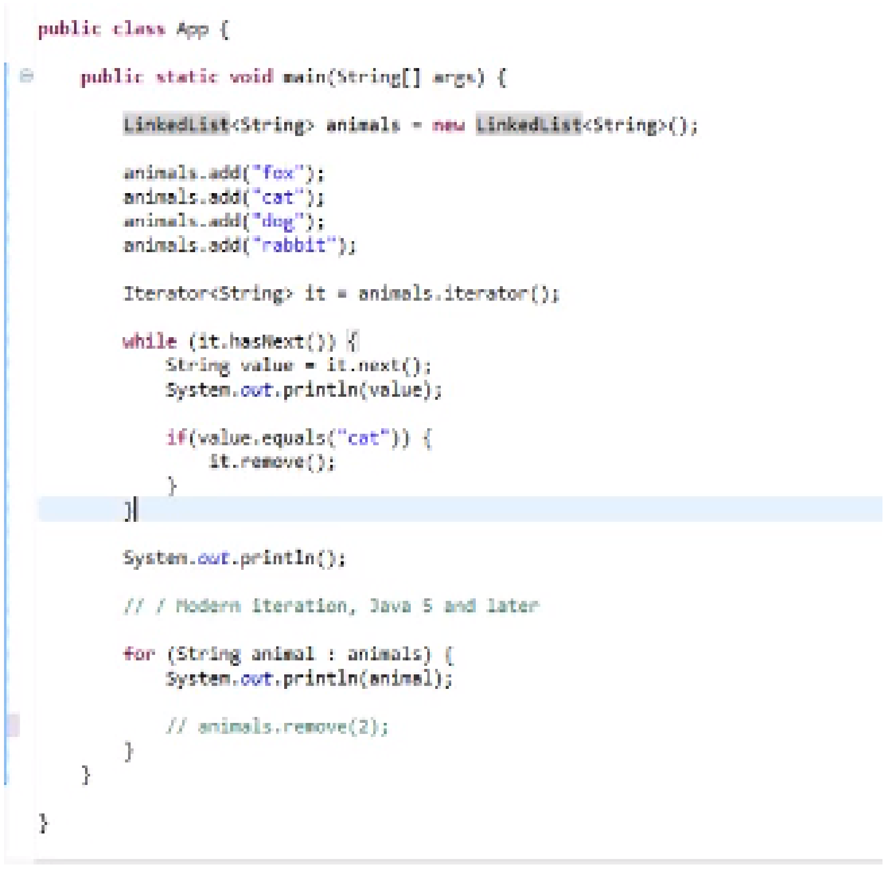
```
class Producer implements Runnable {
  private final BlockingQueue queue;
  Producer(BlockingQueue q) { queue = q; }
  public void run() {
    try {
      while (true) { queue.put(produce()); }
```

Report – Report can be typed or hand written for up to two pages.

The Java collections Frame work:

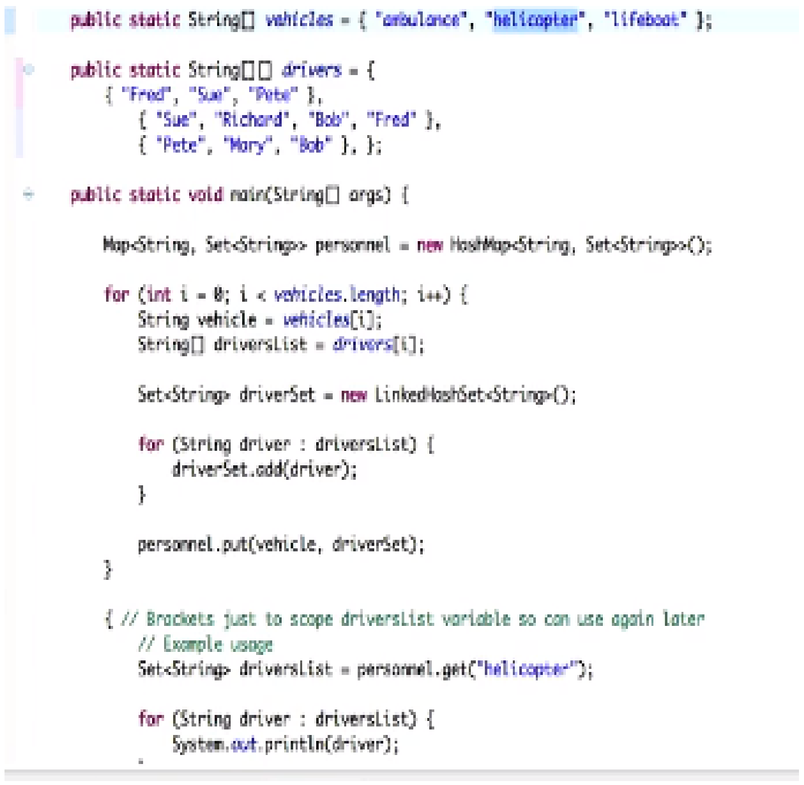
- Natural Ordering
- Queues
- Using Iterators
- Implementing Iterable
- Deciding Which Collection to Use
- Complex Data Structures

Using iterators:



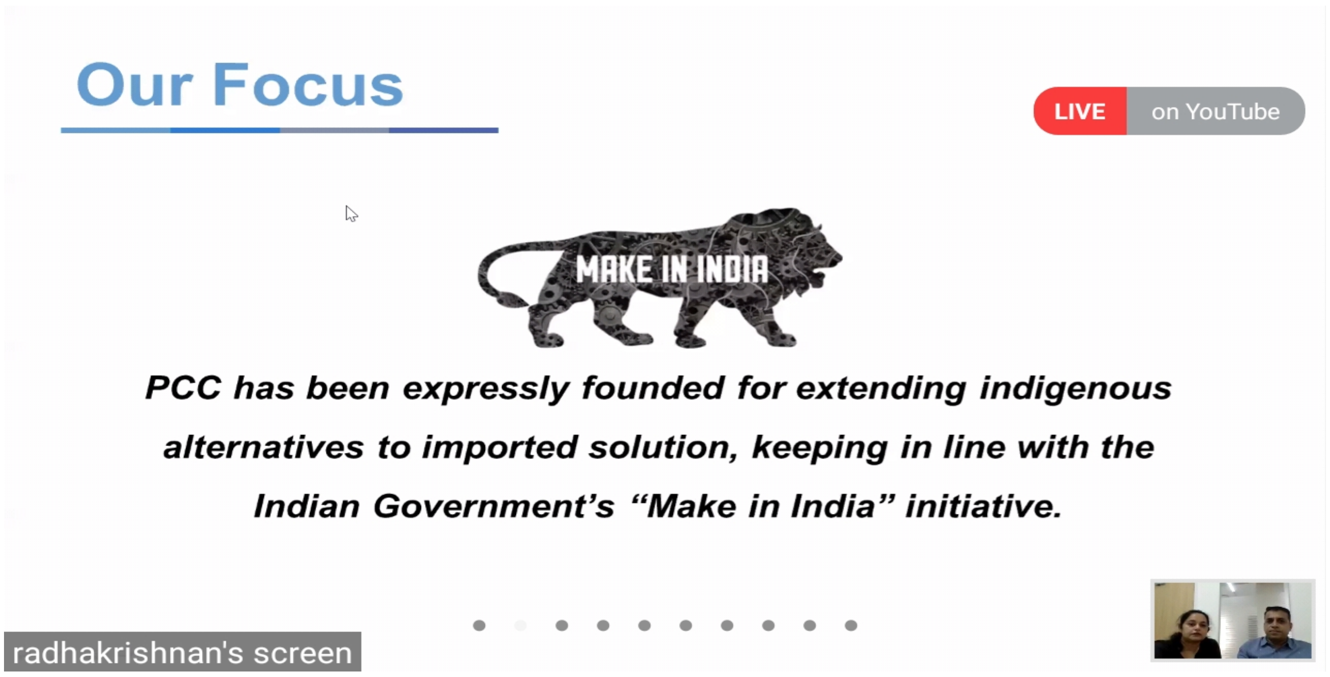
```
public class App {

    public static void main(String[] args) {

        LinkedList<String> animals = new LinkedList<String>();

        animals.add("fox");
        animals.add("cat");
        animals.add("dog");
        animals.add("rabbit");

        Iterator<String> it = animals.iterator();

        while (it.hasNext()) {
            String value = it.next();
            System.out.println(value);

            if(value.equals("cat")) {
                it.remove();
            }
        }

        System.out.println();

        // / Modern iteration, Java 5 and later

        for (String animal : animals) {
            System.out.println(animal);

            // animals.remove(2);
        }
    }

}
```

Complex data structures:



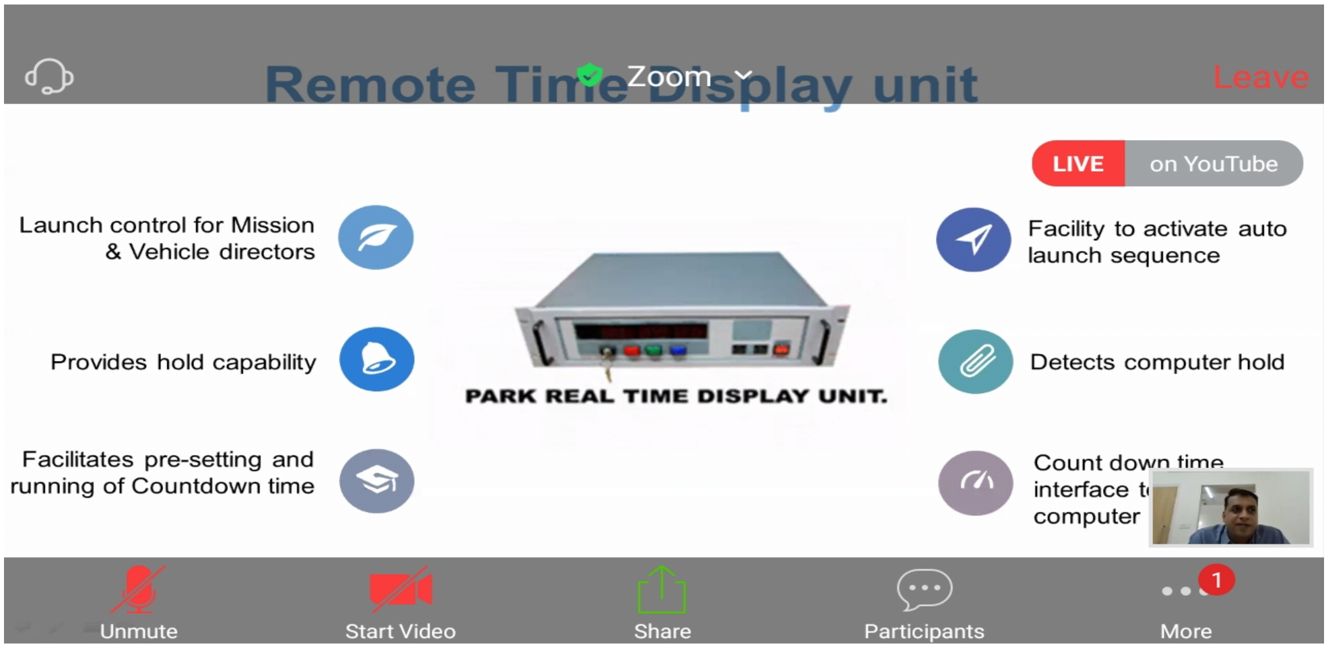
```
public static String[] vehicles = { "ambulance", "helicopter", "lifeboat" };

public static String[][] drivers = {
    { "Fred", "Sue", "Pete" },
        { "Sue", "Richard", "Bob", "Fred" },
        { "Pete", "Mary", "Bob" }, };

public static void main(String[] args) {

    Map<String, Set<String>> personnel = new HashMap<String, Set<String>>();

    for (int i = 0; i < vehicles.length; i++) {
        String vehicle = vehicles[i];
        String[] driversList = drivers[i];

        Set<String> driverSet = new LinkedHashSet<String>();

        for (String driver : driversList) {
            driverSet.add(driver);
        }

        personnel.put(vehicle, driverSet);
    }

    { // Brackets just to scope driversList variable so can use again later
        // Example usage
        Set<String> driversList = personnel.get("helicopter");

        for (String driver : driversList) {
            System.out.println(driver);
```

Date:	16/06/2020	Name:	Akshatha M Deshpande
Course:	Webinar by Mr.RadhaKrishnan M	USN:	4AL17EC006
Topic:	An overview of Avionics in electronics industry	Semester & Section:	6th Sem A sec



Our Focus

PCC has been expressly founded for extending indigenous alternatives to imported solution, keeping in line with the Indian Government's "Make in India" initiative.



Remote Time Display unit

Launch control for Mission & Vehicle directors

Provides hold capability

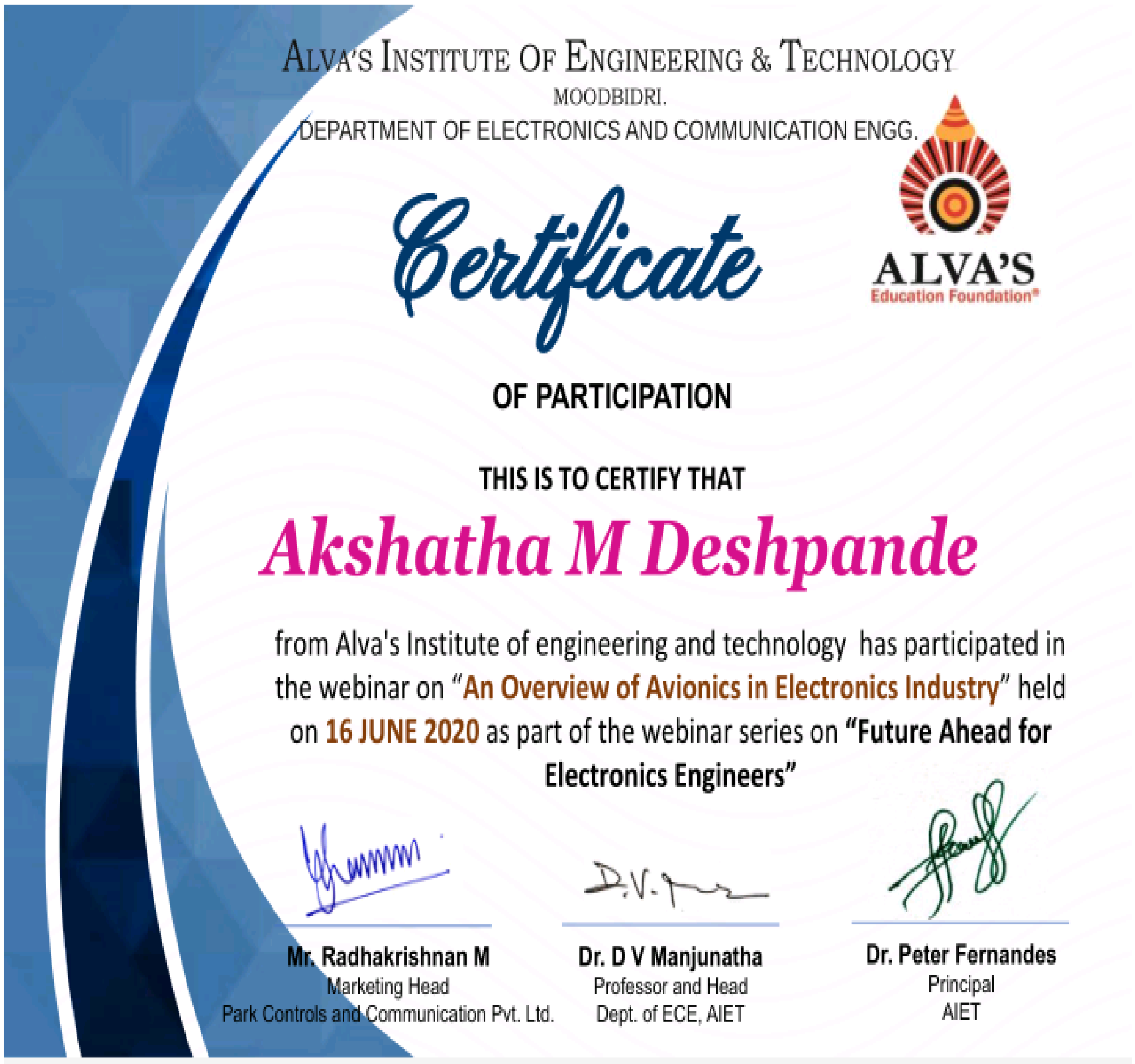
Facilitates pre-setting and running of Countdown time

Facility to activate auto launch sequence

Detects computer hold

Count down time interface t computer

CERTIFICATE:



ALVA'S INSTITUTE OF ENGINEERING & TECHNOLOGY

MOODBIDRI.

DEPARTMENT OF ELECTRONICS AND COMMUNICATION ENGG.

ALVA'S
Education Foundation®

Certificate

OF PARTICIPATION

THIS IS TO CERTIFY THAT

Akshatha M Deshpande

from Alva's Institute of engineering and technology has participated in the webinar on "**An Overview of Avionics in Electronics Industry**" held on **16 JUNE 2020** as part of the webinar series on **"Future Ahead for Electronics Engineers"**

[illegible] [illegible] [illegible]

Mr. Radhakrishnan M
Marketing Head
Park Controls and Communication Pvt. Ltd.

Dr. D V Manjunatha
Professor and Head
Dept. of ECE, AIET

Dr. Peter Fernandes
Principal
AIET