

# DAILY ASSESSMENT FORMAT

Date:	16 <sup>th</sup> June 2020	Name:	Roshni A B
Course:	Statistical Learning	USN:	4AL17EC080
Topic:	Case Study on statistics & probability theory, Solution for case study	Semester & Section:	6 <sup>th</sup> sem 'B' sec
Github Repository:	Roshni-online		

## FORENOON SESSION DETAILS



Edit with WPS Office

## **Agenda:**

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

## **Case study on statistics:**

Case study research is a qualitative research method that is used to examine contemporary real-life situations and apply the findings of the case to the problem under study. Case studies involve a detailed contextual analysis of a limited number of events or conditions and their relationships. It provides the basis for the application of ideas and extension of methods. It helps a researcher to understand a complex issue or object and add strength to what is already known through previous research.

## **STEPS OF CASE STUDY METHOD**

In order to ensure objectivity and clarity, a researcher should adopt a methodical approach to case studies research. The following steps can be followed:

1. Identify and define the research questions - The researcher starts with establishing the focus of the study by identifying the research object and the problem surrounding it. The research object would be a person, a program, an event or an entity.
2. Select the cases - In this step the researcher decides on the number of cases to choose (single or multiple), the type of cases to choose (unique or typical) and the approach to collect, store and analyze the data. This is the design phase of the case study method.
3. Collect the data - The researcher now collects the data with the objective of gathering multiple sources of evidence with reference to the problem under study. This evidence is stored comprehensively and systematically in a format that can be referenced and sorted easily so that converging lines of inquiry and patterns can be uncovered.
4. Evaluate and analyze the data - In this step the researcher makes use of varied methods to analyze qualitative as well as quantitative data. The data is categorized, tabulated and cross checked to address the initial propositions or purpose of the study. Graphic techniques like placing information into arrays, creating matrices of categories, creating flow charts etc. are used to help the investigators to approach the data from different ways and thus avoid making premature conclusions. Multiple investigators may also be used to examine the data so that a wide variety of insights to the available data can be developed.



Edit with WPS Office

**5. Presentation of Results -** The results are presented in a manner that allows the reader to evaluate the findings in the light of the evidence presented in the report. The results are corroborated with sufficient evidence showing that all aspects of the problem have been adequately explored. The newer insights gained and the conflicting propositions that have emerged are suitably highlighted in the report.



Edit with WPS Office

# DAILY ASSESSMENT FORMAT

Date:	16-06-2020	Name:	Roshni A B
Course:	JAVA	USN:	4AL17EC080
Topic:	Queues.iterators.complex data structures.	Semester and section:	6 <sup>th</sup> sem and 'B' sec

\* Queues :-

```

public class App {
    public static void main (String [] args) {
        Queue<Integer> q1 = new ArrayBlockingQueue<Integer>(3);
        q1.add(10);
        q1.add(20);
        q1.add(30);
        System.out.println ("Head of queue is : " + q1.element());
        try {
            q1.add(40);
        } catch (IllegalStateException e) {
            System.out.println ("Tried to add to many items
                                to the queue . . .");
        }
        for (Integer value : q1) {
            System.out.println ("Queue value: " + value);
        }
        System.out.println ("Removed from queue : " + q1.remove());
        System.out.println ("Removed from queue : " + q1.remove());
        System.out.println ("Removed from queue : " + q1.remove());
        try {
            System.out.println ("Removed from queue : " + q1.remove());
        } catch (NoSuchElementException e) {
            System.out.println ("Tried to remove too many items from queue");
        }
        Queue<Integer> q2 = new ArrayBlockingQueue<Integer>(2);
        q2.offer(10);
        q2.offer(20);
        if (q2.offer(30) == false) {
            System.out.println ("Offer failed to add third item . . .");
        }
    }
}

```



```
for (Integer value : q) {  
    System.out.println("Queue 2 value: " + value);  
}
```

3  
3

\* Complex Data Structure :-

```
public class App {  
    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 driverlist = drivers[i];  
            Set<String> driverSet = new LinkedHashSet<String>();  
        }  
    }  
}
```

