Business Intelligence assignment

1) Jio launched publicly in September 2016, where it provided free internet to everyone who wished to acquire a jio connection throughout the country. The tier 1 customers could not be attracted widely towards the offer, whereas the tier 2 and tier 3 customers were highly attracted as the users could easily surf over the internet without thinking about the data consumption or the amount they would need to pay as it was absolutely free of cost and hence the company got a great response from them.

   The video/content applications like Youtube and Tiktok have a very user-friendly platform which helps them to gain and attract more users easily. As on the other hand the online ordering applications have a little more complicated platform as compared to the former.

   The first point could be that the lower tier users are not as smart as the upper tier users and even today they seek the help of others while ordering goods online as it includes some things which are still considered sophisticated such as filling up of forms, the payment portal and also one of the most important thing is the trust that is not yet created as there are many cases where people have been frauded/cheated.

   Online payment has always been problematic to the lower tiers in India. In applications like Youtube and Tiktok, the contents are free to some extent which does not require any online payment process to pay for watching a content. Leaving the tire thing aside we can consider an example which most of us face in our day to day life. Our mothers sees for an item she needs, and then asks us to order it for her as she is not familiar with the technology and also she is not sure that if the product she saw is not a fraud one. On the other hand, but she uses Youtube for watching some cooking videos, listening to songs and many other things which does not require any surety about being fraud.

   Second point could be that the video content applications provide entertainment, knowledge, etc. while the other does not have something of the similar kind in free of cost.

   One other point could be that video content applications use a large amount of data. Before Jio was launched the other data providing companies limited its users with some or the other barrier, but jio removed the data constrain by which the customers were more attracted towards.

   Applications like Flipkart and amazon are not so successful in the rural area as they are in urban areas because in urban areas product are delivered the same day they are ordered or in a span of a week whereas it takes nearly two weeks or even more sometimes to deliver the items.

   Essential commodities hence cannot gain much attraction from the users from rural areas thereby decreasing the number of users in that category as those customers can buy it from the retail stores located nearby. Taking my example- I live in Kalimpong located in West Bengal. It takes about 14-20 days for an order to reach my place , so non-essential items can be ordered but the need for essential item forbids me to do so due to the time constrain that has to be faced.

   The applications like Youtube and Tiktok are some what addictive wherein after you watch a video the application algorithms suggest you a video similar to what you previously watched hereby gathering more attraction towards it.

2) Looking into the data that has been provided:

   The number of new installs grew at the same rate as it was previously growing.

   The first-time buyers doubled which meant that the free shipping for the first-time buyer's idea was a success.

   The fourth column clearly shows that the revenue generated has increased but the average billing per order has decreased.

The first-time buyers who purchased for second time has also increased.

The overall metrics shows that it was an overall success as the revenue earned by the company has increased in regards of the first time buyers but we have not been provided with the data of the existing customers due to which we cannot say that whether some of the first time buyers are the already existing customers who have made a new account to avail the free shipping offer and has abandoned their old accounts. As you can see the ratio of FTBs and FTBs who purchased for a second time was 2:1 but that decreased after July to 5:2 approximately which shows that more customers now only avail the free shipping order. By offering "10000 products under 999" the average billing decreased as more people tend to buy cheaper products.

But when we look overall, we get to know that the company was in profit and hence from that we can conclude that the ideas and plans of the company was a success.

3)

A. Users will spend more time on the application watching the movies and less time searching for one. If the recommendations are good, then user retention will increase, and users will also use the recommendations to watch similar kinds of movies.
B. Goals of the feature should be to increase the watch time of the current users and to attract new users. Metrics for this feature should be movies of the same genre, movies from the same actors, movies from the same directors, etc.
C. Before launching this feature, the company should take a poll from the current users if they would want this feature in the application so that the users will get the knowledge of what this company is planning to bring up for the users. And after some time when the feature is ready to roll, an in-app popup and the carousel with the new recommendations with the home screen should be displayed. This feature could be evaluated by the increase in the watch time by users or by taking a quick feedback through rating by the customers.

Android Assignment

2.Let’s see the concurrent modification exception scenario with an example.

```
package com.journaldev.ConcurrentModificationException;
import java.util.ArrayList;
import java.util.HashMap;
import java.util.Iterator;
import java.util.List;
import java.util.Map;

public class ConcurrentModificationExceptionExample {
	public static void main(String args[]) {
		List<String> myList = new ArrayList<String>();
		myList.add("1");
		myList.add("2");
		myList.add("3");
		myList.add("4");
		myList.add("5");
		Iterator<String> it = myList.iterator();
		while (it.hasNext()) {
			String value = it.next();
			System.out.println("List Value:" + value);
			if (value.equals("3"))
				myList.remove(value);
		}
		Map<String, String> myMap = new HashMap<String, String>();
		myMap.put("1", "1");
		myMap.put("2", "2");
		myMap.put("3", "3");
		Iterator<String> it1 = myMap.keySet().iterator();
		while (it1.hasNext()) {
			String key = it1.next();
			System.out.println("Map Value:" + myMap.get(key));
			if (key.equals("2")) {
				myMap.put("1", "4");
				// myMap.put("4", "4");
			}
		}

	}
}
```

Above program will throw java.util.ConcurrentModificationException when executed
The following methods can be used to avoid ConcurrentModificationException:

1. You can convert the list to an array and then iterate on the array. This approach works well for small or medium size list but if the list is large then it will affect the performance a lot.
2. You can lock the list while iterating by putting it in a synchronized block. This approach is not recommended because it will cease the benefits of multithreading.
3. If you are using JDK1.5 or higher then you can use **ConcurrentHashMap** and **CopyOnWriteArrayList** classes. This is the recommended approach to avoid concurrent modification exception.
4. You can use the iterator remove() function to remove the object from underlying collection object. But in this case, you can remove the same object and not any other object from the list.

Let's run an example using Concurrent Collection classes.

```
package com.journaldev.ConcurrentModificationException;

import java.util.Iterator;
import java.util.List;
import java.util.Map;
import java.util.concurrent.ConcurrentHashMap;
import java.util.concurrent.OnWriteArrayList;

public class AvoidConcurrentModificationException {

public static void main(String[] args) {

	List<String> myList = new OnWriteArrayList<String>();

	myList.add("1");
	myList.add("2");
	myList.add("3");
	myList.add("4");
	myList.add("5");

	Iterator<String> it = myList.iterator();
	while (it.hasNext()) {
		String value = it.next();
		System.out.println("List Value:" + value);
		if (value.equals("3")) {
			myList.remove("4");
			myList.add("6");
			myList.add("7");
		}
	}
	System.out.println("List Size:" + myList.size());
```

```
		Map<String, String> myMap = new ConcurrentHashMap<String, String>();
		myMap.put("1", "1");
		myMap.put("2", "2");
		myMap.put("3", "3");

		Iterator<String> it1 = myMap.keySet().iterator();
		while (it1.hasNext()) {
			String key = it1.next();
			System.out.println("Map Value:" + myMap.get(key));
			if (key.equals("1")) {
				myMap.remove("3");
				myMap.put("4", "4");
				myMap.put("5", "5");
			}
		}

		System.out.println("Map Size:" + myMap.size());
	}

}
```

The output of the above program is shown below. You can see that there is no ConcurrentModificationException being thrown by the program

```
List Value:1
List Value:2
List Value:3
List Value:4
List Value:5
List Size:6
Map Value:1
Map Value:2
Map Value:4
Map Value:5
Map Size:4
```

From the above example it's clear that:

1. Concurrent Collection classes can be modified safely, they will not throw ConcurrentModificationException.
2. In case of CopyOnWriteArrayList, iterator doesn't accommodate the changes in the list and works on the original list.
3. In case of ConcurrentHashMap, the behaviour is not always the same.
4. Use for loop to avoid java.util.ConcurrentModificationException

If you are working on single-threaded environment and want your code to take care of the extra

added objects in the list then you can do so using for loop rather than an Iterator.

```
        for(int i = 0; i<myList.size(); i++){
                System.out.println(myList.get(i));
                if(myList.get(i).equals("3")){
                myList.remove(i);
                i--;
                myList.add("6");
        }
}
```