OBJECTIVES:
1.ArithmaticEXception(B)
2.start exception finally end(B)
3.car maruti tata(A)
4.MyException(A)
5.cannat create instance of child class(D)
6.2 3 4 5(D)
7.range(D)
8.tr:nth-child(even) {background-color:#f2f2f2;}(A)
9.init,service,destroy;(B)
10.mouseover(C)

SUBJECTIVE:
1.What is the difference between array list and linked list provide the senario’s for each one’s usage?

->ArrayList provides constant time for search operation.

```
    Eg:List lst = new ArrayList<>(2);
               lst.add(0);
               lst.add(1);
               lst.add(2);
               lst.add(1);
               lst.add(4);
               lst.add(5);
             System.out.println(lst);
            o/p: [0,1,2,1,4,5]
```

->The LinkedList provides constant time for add and remove operations.

```
    Eg: Queue<Integer> q= new LinkedList<>();
        for (int i = 0; i < 5; i++)
        q.add(i);
        System.out.println("Elements of queue "+ q);
```

2.What is the error page how to create one in jsp?

Error pages are the result of an HTTP status code. This status code is the server’s response – in a three-digit code – to your browser’s request. Meaning that when you type a site’s URL in your browser, it sends a request to the server. What it receives is a code – an HTTP status code.

To create a JSP error page, we need to set page directive attribute isErrorPage value to true, then we can access exception jsp implicit object in the JSP and use it to send customized error message to the client.

```
Eg:<html>
   <body>
<%@ page isErrorPage="true" %>
error:
<%= exception %>
</body>
</html>
```

3.What are the different types of tags available in jsp.

There are Three types of tags are there:-

->scriptlet tag
->expression tag

->declaration tag

4.Why HTML is stateless.

HTTP is called as a stateless protocol because each request is executed independently, without any knowledge of the requests that were executed before it, which means once the transaction ends the connection between the browser and the server is also lost.

5.What are the ways to iterate overlist?What is the fail fast case in collections?How to ovwecome the same?

1.Simple For loop.
2.Enhanced For loop.
3.Iterator.
4.ListIterator.
5.While loop.
6.Iterable.forEach() util.
7.Stream.forEach() util.

The Fail Fast iterators immediately throw ConcurrentModificationException in case of structural modification of the collection. Structural modification means adding, removing, updating the value of an element in a data collection while another thread is iterating over that collection.

OverCome:-

The Fail Safe iterators are just opposite to Fail Fast iterators; unlike them, A fail-safe iterator does not throw any exceptions unless it can handle if the collection is modified during the iteration process. This can be done because they operate on the copy of the collection object instead of the original object. The structural changes performed on the original collection ignored by them and affect the copied collection

6.What is the difference between HttpSession's getSession(),getSession(true) and getSession(false) methods which one is preferred?

The methods getSession() and getSession(boolean) are very similar. There's a small difference, though. The difference is whether the session should be created if it doesn't exist already.

Calling getSession() and getSession(true) are functionally the same: retrieve the current session, and if one doesn't exist yet, create it.I would prefer Calling getSession(false), though, retrieves the current session, and if one doesn't exist yet, returns null. Among other things, this is handy when we want to ask if the session exists.

7.What is the memory leakage?How to prevent the same.

Memory leak occurs when programmers create a memory in heap and forget to delete it.

The consequences of memory leak is that it reduces the performance of the computer by reducing the amount of available memory. Eventually, in the worst case, too much of the available memory may become allocated and all or part of the system or device stops working correctly, the application fails, or the system slows down vastly .

To prevent

1. Copy objects instead of passing references. Pass a reference only if the object is huge and a copy operation is expensive.
2. Avoid object mutations as much as possible. ...
3. Avoid creating multiple references to the same object. ...
4. Use short-lived variables.
5. Avoid creating huge object trees.

8. Can we declare class as final?

You can declare any class as final, but you cannot extend it.

9. How can we ensure that a resource is not used by multiple threads simultaneously in multi threading.

In multi-threading, access to the resources which are shared among multiple threads can be controlled by using the concept of synchronization. Using synchronized keyword, we can ensure that only one thread can use shared resource at a time and others can get control of the resource only once it has become free from the other one using it.

10. What is cloning?Where is the cloning used?

1ST PROGRAM:

```
import java.util.Scanner;

public class Sum_of_two_no {

    public static void main(String[] args) {
        Scanner sc=new Scanner(System.in);
        System.out.print("Enter the Size of Array : ");
        int num = sc.nextInt();
        int arr[] = new int[num];
        for(int i=0; i<num; i++)
                arr[i] = sc.nextInt();

        System.out.println("Enter the number ");
        int n = sc.nextInt();
        int flag=0;
        for(int i=0; i<num; i++)
        {
            for(int j  = 0; j<num; j++)
            {
                if(arr[i]+arr[j] == n) {
                    flag++;
                    System.out.println("{"+arr[i]+","+arr[j]+"}");

                }
            }
        }
        if(flag==0)
            System.out.println("can't get value for given sum ");

    }

}
```

2nd PROGRAM:

```
import java.util.*;
import java.io.*;

class test{
    static int MAX = 256;
    private boolean checkString(String str1, String str2)
    {
        int[] count = new int[MAX];
        char []str3 = str1.toCharArray();
        for (int i = 0; i < str3.length; i++)
            count[str3[i]]++;

        char []str4 = str2.toCharArray();
        for (int i = 0; i < str4.length; i++) {
            if (count[str4[i]] == 0)
                return false;
            count[str4[i]]--;
        }
        return true;
    }
    public static void main(String args[])
    {
        Scanner sc =  new Scanner(System.in);
        System.out.println("Enter First String : ");
        String str1 = sc.nextLine();
        System.out.println("Enter Second String : ");
        String str2 = sc.nextLine();
        test obj = new test();
        boolean flag = obj.checkString(str1,str2);
        System.out.println(flag);
    }
}
```