PROGRAMMING IN JAVA

1.
```java
import java.util.*;

class power_of_three {
    public static boolean ispower(int n) {
        if(n < 3)
        return false;
        while(n % 3 == 0) {
                n = n / 3;
            }
        return n == 1;
    }
    public static void main(String args[]) {
        Scanner s = new Scanner(System.in);
        System.out.println("enter n:");
        int n = s.nextInt();
        if(ispower(n)) {
            System.out.println(n + " is power of 3");
        }
        else
        System.out.println(n + " is not power of 3");
    }
}
```

2.
```java
import java.util.*;

class MostFrequentWord {
    public static String MostCommonWord(String para,String banned[]) {
        Set<String> bannedset = new HashSet<>(Arrays.asList(banned));
        Map<String,Integer> wordFreq = new HashMap<>();
        String words[] = para.replaceAll("[a-zA-
Z]","").toLowerCase().split("\\s+");
        for(String word:words) {
            if(!bannedset.contains(word)) {
                wordFreq.put(word,wordFreq.getOrDefault(word,0)+1);
            }
        }
        String MostFrequentWord = null;
        int maxFreq = 0;
        for(Map.Entry<String,Integer> entry:wordFreq.entrySet()) {
            if(entry.getValue() > maxFreq) {
                MostFrequentWord = entry.getKey();
                maxFreq = entry.getValue();
            }
        }
```

```
        return MostFrequentWord;
    }
    public static void main(String args[]) {
        String para = "Ram hit a ball, the hit ball flew far after it was
hit";
        String banned[] = {"ball"};
        String result = MostCommonWord(para,banned);
        System.out.println(result);
    }
}
```

3.
```
import java.util.*;

class Missing_number {
    public static void main(String args[]) {
        int a[] = {3,0,1};
        int i,n,miss,sum=0,sum1=0;
        Scanner s = new Scanner(System.in);
        System.out.println("enter n:");
        n = s.nextInt();
        sum = (n * (n+1)) / 2;
        for(i=0;i<n;i++) {
            sum1 = sum + a[i];
        }
        System.out.println("Missing number:"+(sum - sum1));
    }
}
```

4.
```
class thread1 implements Runnable {

    public void run() {
        System.out.println("thread1 started");
        try {
            Thread.sleep(1000);
        }
        catch(Exception e) {
            System.out.println(e);
        }
        System.out.println("thread1 finished");
    }
}
class thread2 implements Runnable {
    public void run() {
        System.out.println("thread2 started:");
        try {
            Thread.sleep(1500);
        }
        catch(Exception e)
        {
            System.out.println(e);
```

```
        }
        System.out.println("thread1 finished");
    }
}
class day6Aq6 {
    public static void main(String args[]) {
        Thread t1 = new Thread(new thread1());
        Thread t2 = new Thread(new thread2());
        try {
        t1.start();
        Thread.sleep(500);
        t2.start();
        t1.join();
        t2.join();
        System.out.println("All threads are finished");
        }
        catch(Exception e) {
            System.out.println(e);
        }
    }
}.
```

5.

```
import java.util.*;

class day6Aq7 {
    public static void main(String args[]) {
        Scanner s = new Scanner(System.in);
        int choice;
        while(true) {
            System.out.println(".........Syntax generator for.............");
            System.out.println("1.if");
            System.out.println("2.switch");
            System.out.println("3.while");
            System.out.println("4.do-while");
            System.out.println("5.for");
            System.out.println("6.exit");
            System.out.println("choos choice:");
            choice = s.nextInt();
            System.out.println("");
            switch(choice) {
                case 1:
                System.out.println("the if");
                System.out.println("if(condition) statement:");
                System.out.println("else statement");
                break;
                case 2:
                System.out.println("switch");
                System.out.println("switch(expression) {");
                System.out.println("case constant:");
```

```
                System.out.println("statemenr sequence");
                System.out.println("break");
                System.out.println("}");
                break;
                case 3:
                System.out.println("while");
                System.out.println("while condition statement");
                break;
                case 4:
                System.out.println("do-while");
                System.out.println("do {");
                System.out.println("statment");
                System.out.println("} while (condition)");
                break;
                case 5:
                System.out.println("for");
                System.out.println("for(init:condition:iteration)");
                System.out.println("statement");
                break;
                default:
                System.out.println("enter correct choice:");
                break;
            }
        }
    }
}
```

6.
```
class superclass {

    int x;
    public superclass(int x) {
        this.x = x;
    }
     public void display()
     {
         System.out.println("value of X in super class:"+x);
     }
    }
    class subclass extends superclass {
        int y;
        public subclass(int x,int y) {
             super(x);
              this.x = x;
              this.y = y;
        }
        public void display() {
            System.out.println("value of X  & Y in sub class:"+x +","+ y);
        }
    }
```

```
    class day6p5 {
        public static void main(String args[]) {
            subclass s = new subclass(100,200);
            s.display();
        }
    }
```

8.
```
 class overload {

    public void print(int a) {
        System.out.println("a = "+a);
    }
    public void print(int a,int b) {
        System.out.println("a = " + a + "," + b +"="+"b");
    }
    public static void main(String args[]) {
        overload o = new overload();
        o.print(7);
        o.print(7,9);
    }
```

9.
```
 import java.util.*;


public class CountDistinctCharacters {
    public static void main(String[] args) {
        String str = "hello world";
        Map<Character, Integer> charCountMap = new HashMap<>();

        // iterate through the characters of the string
        for (char c : str.toCharArray()) {
            // check if the character is already in the map
            if (charCountMap.containsKey(c)) {
                // if yes, increment its count by 1
                charCountMap.put(c, charCountMap.get(c) + 1);
            } else {
                // if no, add the character to the map with count 1
                charCountMap.put(c, 1);
            }
        }

        // print the distinct characters and their count
        for (Map.Entry<Character, Integer> entry : charCountMap.entrySet()) {
            System.out.println(entry.getKey() + " : " + entry.getValue());
 }
}
}
```

10.
```
 import java.util.*;
```

```
class Unique_char {
    public static void main(String args[]) {
        Scanner s = new Scanner(System.in);
        System.out.println("enter string:");
        String str = s.nextLine();
        Set<Character> unique = new  HashSet<>();
        Set<Character> nonunique = new HashSet<>();
        for(char c:str.toCharArray()) {
            if(nonunique.contains(c)) {
                continue;
            }
            if(unique.contains(c)) {
                unique.remove(c);
                nonunique.add(c);
            }
            else {
                unique.add(c);
            }
        }
        System.out.println("Unique Characters in the String:");
        for(char c:unique) {
            System.out.print(c+" ");
        }
    }
}
```

11. .

```
class thread1 extends Thread {

    public void run() {
        for(int i=1;i<=5;i++) {
            System.out.println(i+"*"+5+"="+(i*2));
        }
    }
}
class thread2 extends Thread {
    public void run() {
        for(int i=1;i<=10;i++) {
            System.out.println(i+"*"+10+"="+(i*10));
        }
    }
}
class A3q2 {
    public static void main(String args[]) {
        thread1 t1 = new thread1();
        thread2 t2 = new thread2();
        t1.setPriority(Thread.MAX_PRIORITY);
        t1.start();
        t2.start();
        }
```

```
}
```