1)WAJP Print Strong Number Between the Range(1 to 100)

```
package Final_Test;


public class Strong_Number {


	public static void main(String[] args) {
		System.out.println("Strong numbers are:");
		for(int n1=1;n1<=1000;n1++)
		{
			int n=n1;
			int sum=0;
		while(n!=0)
		{
			int fact=1;


			int rem=n%10;
			for(int i=1;i<=rem;i++)
		{
			fact=fact*i;


		}
			sum=sum+fact;
			n=n/10;


		}
		if(sum==n1)
		{
		System.out.println(n1);
```

```
                }
                else
                {

                }
                }
        }

}
```

2)WAJP Print Twisted Prime number Between the Range(1 to 1000)

```
package Final_Test;

public class Twisted_Prime {

        public static void main(String[] args) {
                for(int n1=1;n1<=100;n1++)
                {
                        int c=0,n=n1;



                for(int i=1;i<=n;i++)
                {
                        if(n%i==0)
                        {
                                c++;
                        }
                }
```

```
            if(c==2)
            {
                int rev=0,temp=n;
                while(n>0)
                {

                int rem=n%10;
                rev=(rev*10)+rem;
                n=n/10;
            }



        c=0;
        for(int j=1;j<=rev;j++)
        {
            if(rev%j==0)
            {
                c++;
            }
        }

            if(c==2)
            {
                System.out.println(n1);
            }
        }
}
```

```
	}


}
```

3)WAJP Print Below pattern

```
package Final_Test;


public class Star_pattern {


	public static void main(String[] args) {
		int n=4;
		for(int i=1;i<=n;i++)
		{
			for(int j=1;j<=n*3+1;j++)
			{
				if((i+j>=n+1&&i+j<=n*2)||(j-i>=n+2&&j-i<=n*2+1))
				{
					System.out.print("*  ");
				}
				else
				{
					System.out.print("  ");
				}
			}
			System.out.println();
		}


	}
```

}

4)Print Diamond pattern A B B B C C C C C D D D D D D D C C C C C B B B A

```
package Final_Test;

public class Daimond_Pattern {

	public static void main(String[] args) {
		char ch='A';
		int n=4,ch1='D';
		System.out.println("---Diamond---");
		for(int i=1;i<=n;i++)
		{
			for(int j=1;j<=n*2-1;j++)
			{

				if(i+j>=n+1&&j-i<=n-1&&i+j<=n*3-1&&i-j<=n-1)
				{

					System.out.print(ch+" ");

				}

				else
				{
					System.out.print("  ");
				}
			}
```

```
                System.out.println();
                ch++;
            }
            for(int i=n+1;i<=n*2-1;i++)
            {
                ch1=ch1-1;

                for(int j=1;j<=n*2-1;j++)
                {

                    if(i+j>=n+1&&j-i<=n-1&&i+j<=n*3-1&&i-j<=n-1)
                    {
                            System.out.print((char)ch1+" ");

                        }

                    else
                    {
                        System.out.print("  ");
                    }
                }
                System.out.println();
            }

        }

}
```

5)WAJP Right Rotation?

```
package Final_Test;

import java.util.Arrays;

public class Right_Rotation {

	public static void main(String[] args) {
		int[] a= {1,2,3,4,5};
		int n=5;
		while(n!=0)
		{
			int temp=a[a.length-1];


		for(int i=a.length-1;i>=0;i--)
		{
			if(i!=0)
			{
			a[i]=a[i-1];
		}
			else
			{
		a[i]=temp;
			}

	}

		System.out.println(Arrays.toString(a));
```

```
                n--;
        }

        }

}
```

6)WAJP Fina largest element in row

```
package Final_Test;

import java.util.Scanner;

public class Largest_Element_in_Row {

        public static void main(String[] args) {
                Scanner s=new Scanner(System.in);
                System.out.println("Enter row size");
                int row=s.nextInt();
                System.out.println("Enter col size");
                int col=s.nextInt();
                int[][] a=new int[row][col];

                System.out.println("Enter array elements");
                for(int i=0;i<row;i++)
                {
                        for(int j=0;j<col;j++)
                        {
                                a[i][j]=s.nextInt();
                        }
```

```
                }
                System.out.println("Array Elements are:" );
                for(int i=0;i<row;i++)
                {
                        for(int j=0;j<col;j++)
                        {

                                System.out.print(a[i][j]+" ");

                        }
                        System.out.println();
}

                for(int i=0;i<row;i++)
                {
                        for(int j=0;j<col;j++)
                        {
                                for(int k=0;k<row;k++)
                                {
                                        int max=a[i][j];

                                        for(int l=1;l<col;l++)
                                        {

                                if(max<a[k][l])
                                {
                                        max=a[k][l];
```

```
                }
            }
                    System.out.println("largest element:"+max);
                    i++;

        }
            }
        }



    }

}
```

7)WAJP Reverse each word , if it contain t latter or else print as it is

```
package Final_Test;

public class Reverse_each_word {

    public static void main(String[] args) {
        String s="ram sat cat jerry";
        String[] s1=s.split(" ");
        String s2="";
        for(int i=0;i<s1.length;i++)
        {
            char[] a=s1[i].toCharArray();
            boolean flag=false;
            for(int j=0;j<=a.length-1;j++)
            {
```

```
            if(a[j]=='t')
            {
                flag=true;
            }
        }
        if(flag)
        {
            for(int k=a.length-1;k>=0;k--)
            {
                s2=s2+a[k];
            }

        }
        else
        {
            for(int k=0;k<a.length;k++)
            {
            s2=s2+a[k];
            }
        }
        s2=s2+" ";
    }



        System.out.println(s2);
```

```
	}

}
```

8)WAJP Print Anagram words From String Array

```
package Final_Test;

import java.util.Arrays;

public class Anagram {

	public static void main(String[] args) {
		String s="hello world race care olleh";
		String[] s1=s.split(" ");
		String s2="";
		for(int i=0;i<s1.length;i++)
		{
			for(int j=i+1;j<s1.length;j++)
			{
			char[] a=s1[i].toCharArray();
			char[] b=s1[j].toCharArray();

			Arrays.sort(a);
			Arrays.sort(b);

		String s3=new String(a);
		String s4=new String(b);
		if(s3.equals(s4))
```

```
                {
                        System.out.println(s1[i]);

                }

                }
                }
                        }

}
```