

Lista de enlace simple (Con un dato estructurado)

No es nada más que la lista que ya sabemos utilizar, simplemente que cambiaremos el tipo de dato de nuestra información de nuestra clase nodo.

La información que hasta ahora se ha utilizado en nuestro nodo han sido tipos de datos primitivos(int, float, char, boolean, byte, short para nuestro caso en java), hoy les dejo un ejemplo de cómo hacer el manejo de las listas con tipos de datos definidos por el usuario.

Clase Estudiante.- Es la clase (tipo de dato definido por el usuario) que utilizaremos como información de nuestra clase nodo.

```
public class Estudiante{
    String nombre;
    int edad;
    String CI;
    public Estudiante(String n,int e,String c) {
        nombre=n;
        edad=e;
        CI=c;
    }
    @Override
    public String toString(){
        return nombre+"\t"+edad+"\t"+CI+"\n";
    }
}
```

La función *toString()*, esta clase nos ayudara en la impresión de los datos del estudiante.

Clase Nodo.- En esta clase utilizamos la clase *Estudiante* como parte de la información de nuestro nodo.

```
public class Nodo {
    Estudiante info;
    Nodo enla;
    public Nodo(Estudiante x) {
        info=x;
        enla=null;
    }
}
```



```
        tem=tem.enla;
    }
}
return con;
}
float promedio() {
    int sum=0, con=0;
    if(cab!=null) {
        Nodo tem=cab;
        while(tem!=null) {
            sum+=tem.info.edad;
            con++;
            tem=tem.enla;
        }
    }
    return (float) sum/con;
}
Estudiante mayor() {
    Estudiante e=null;
    int may=0;
    if(cab!=null) {
        Nodo tem=cab;
        while(tem!=null) {
            if(tem.info.edad>may) {
                may=tem.info.edad;
                e=tem.info;
            }
            tem=tem.enla;
        }
    }
    return e;
}
void InsertarNoRepetido(Estudiante x) {
    if(cab==null) {
        cab=new Nodo(x);
    }
    else{
        Nodo p=cab;
        while(p!=null) {
            if(p.info.CI.compareTo(x.CI)==0)
                break;
            p=p.enla;
        }
        if(p==null) {
            Nodo tem=new Nodo(x);
            tem.enla=cab;
            cab=tem;
        }
    }
}

void Intercalar(Nodo x) {
    if(cab==null)
        cab=x;
    if(x!=null) {
        Nodo tem1=cab, tem2=cab.enla;
        while(x!=null&&tem2!=null) {
            tem1.enla=x;
        }
    }
}
```

```
        tem1=x;
        x=x.enla;
        tem1.enla=tem2;
        tem1=tem2;
        tem2=tem2.enla;
    }
    if(x!=null) {
        tem1.enla=x;
    }
}
}
```

La prueba esta hecha para introducir a 5 estudiantes.

El programa solo permite un nombre por estudiante.

```
import java.util.Scanner;

public class Prueba {

    public static void main(String[] args) {
        Lista lis=new Lista();
        Lista l=new Lista();
        Estudiante est;
        int n=1;
        Scanner cs=new Scanner(System.in);
        String nom;
        int eda;
        String ci;
        while(n<6) {
            System.out.println("introduzca el nombre del
estudiante\t"+n);
            nom=cs.next();
            System.out.println("introduzca la edad del
estudiante\t"+n);
            eda=cs.nextInt();
            System.out.println("introduzca el CI del
estudiante\t"+n);
            ci=cs.next();
            est=new Estudiante(nom, eda, ci);
            lis.insertar(est);
            l.InsertarNoRepetido(est);
            n++;
        }
        System.out.println("primera lista");
        lis.mostrar();
        System.out.println("segunda lista");
        l.mostrar();
        System.out.println("intercalados");
        lis.Intercalar(l.cab);
        lis.mostrar();
    }
}
```