

**NOMBRES:** ALBERTO ERICK TEJEDA  
SÁNCHEZ

MARIA JOSE COSME ESTRADA

**MATERIA**

ESTRUCTURA DE DATOS

**GRUPO**

3G2D

**PROFESOR:** RAFAEL HERRERA GARCIA

**CARRERA:** INGENIERÍA EN SISTEMAS  
COMPUTACIONALES

# CODIGO

```
package estable; import
```

```
java.util.Scanner;
```

```
public class Ejemplar {
```

```
    private int codigorp, edad, año ;    private String  
fechaAd, raza ; // fecha rl registro de leche    private float  
cleche;  
    int fr;
```

```
    public Ejemplar(int codigorp, int edad, int año, String fechaAd, String raza, float cleche, int fr) {  
this.codigorp = codigorp;    this.edad = edad;    this.año = año;    this.fechaAd = fechaAd;  
    this.raza = raza;  
this.cleche = cleche;  
    this.fr = fr;  
    }
```

```
    public int getCodigorp() {  
return codigorp;  
  
    }
```

```
    public int getEdad() {  
return edad;  
    }
```

```
    public int getAño() {  
return año;  
    }
```

```
    public String getFechaAd() {  
return fechaAd;  
    }
```

```
    public String getRaza() {  
return raza;  
    }
```

```
    public float getCleche() {  
return cleche;  
    }
```

```
    public void setCodigorp(int codigorp) {  
this.codigorp = codigorp;  
    }
```

```
    public void setEdad(int edad) {  
this.edad = edad;  
    }
```

```
    public void setAño(int año) {  
this.año = año;  
    }
```

```
    public void setFechaAd(String fechaAd) {  
this.fechaAd = fechaAd;  
    }
```

```
    public void setRaza(String raza) {  
        this.raza = raza;  
    }
```

```
    public void setCleche(float cleche) {  
this.cleche = cleche;  
    }
```

```
    public int getFr() {  
return fr;  
    }
```

```
    public void setFr(int fr) {  
        this.fr = fr;  
    }
```

```
package estable;
```

```
import java.io.FileInputStream; import
java.io.FileNotFoundException; import
java.util.Scanner;
```

```
public class ejemplares {
```

```
    final int N=500;
```

```
        private int limite;
```

```
private Ejemplar lista[];
```

```
    public ejemplares() throws FileNotFoundException {
```

```
        limite=0;
```

```
        lista= new Ejemplar[N];
```

```
cargaEjemplares();
```

```
    }
```

```
private void cargaEjemplares() throws FileNotFoundException {
```

```
    Scanner input;
```

```
        input = new Scanner(new FileInputStream("granja.txt"));
```

```
        input.useDelimiter(";");
```

```
        while(input.hasNext()) {
```

```

        int c=input.nextInt(); //codigorp
int e=input.nextInt();//edad                int
a=input.nextInt();//año

        String ad=input.nextLine(); //fecha adquisicion
String r=input.nextLine(); //raza

        float l=input.nextFloat(); //litros

        int fr=input.nextInt();//fecha registro


//int c, int e, int a, int ad, String raza, FLOAT L String s


        lista[limite++]= new Ejemplar (c,e,a,ad,r,l,fr);

    }

    input.close();

}

private float lechePromedio() {
    float promedio=0;
    for(int i=0;i<limite;i++)
        promedio+=lista[i].getCleche();
return promedio/limite;
}

private float Maxproduccion() {
    float mp=0;

```

```

        for(int i=0;i<limite;i++)
            if(lista[i].getCleche(>mp)
                mp=lista[i].getCodigorp());
        return mp;
    }

}

public void imprimeDatos () {
    for(int i=0;i<limite;i++) {
        System.out.print("codigo de registro"+lista[i].getCodigorp());
        System.out.print("\t edad "+lista[i].getEdad());
        System.out.print("\t año "+lista[i].getAño());
        System.out.print("\t año de adquisicion "+lista[i].getFechaAd());
        System.out.println("\t raza"+lista[i].getRaza());
        System.out.print("\t cantidad leche "+lista[i].getCleche());
        System.out.println("\t mes"+lista[i].getFr() );

    }
}

public void imprimePromedio() {
    System.out.println("El sueldo promedio="+lechePromedio());
}

```

```
}
```

```
public void imprimeMaxproduccion() {  
    float mprod=Maxproduccion();  
    System.out.println("El valor mayor de produccion="+mprod);  
    System.out.println("Los codigos con mayor produccion son:");  
    for(int i=0;i<limite;i++)  
        if(lista[i].getCleche()==mprod)  
            System.out.println(lista[i].getCodigorp());  
}
```

```
}
```

```
}
```

```
import java.io.FileNotFoundException; import  
java.util.Scanner;
```

```
public class Establo {
```



```
public static void main(String[] args) throws FileNotFoundException {  
    // TODO Auto-generated method stub  
  
    ejemplares e = new ejemplares();  
  
        e.imprimeDatos();  
    e.imprimeMaxproduccion();  
    e.imprimePromedio();  
}  
  
}
```

#### Observaciones

Observamos que este problema es muy similar al visto en clase solo que, con otras implicaciones, no vimos necesario el uso de la herencia ni polimorfismo pues prácticamente todos los atributos eran lo mismo y no había un método con un procedimiento especial para cada caso que haya sido otra cosa es que el archivo de texto a veces da problemas por los espacios vacíos en los tipos de dato int o flotante por lo que los errores en la mayoría de ocasiones provenían solo del archivo de texto

#### Referencias

Nos basamos en el ejercicio visto en clase pudimos hacerlo de una forma distinta con la creación de un archivo de texto pero implicaba meter datos por teclado por lo que optamos por algo parecido al de la clase, también nos basamos en como estaba conformado el archivo de texto para su utilización

LINK DEL REPOSITORIO [https://github.com/albertotejeda21/EJERCICIOTDA\\_PRAC1](https://github.com/albertotejeda21/EJERCICIOTDA_PRAC1)