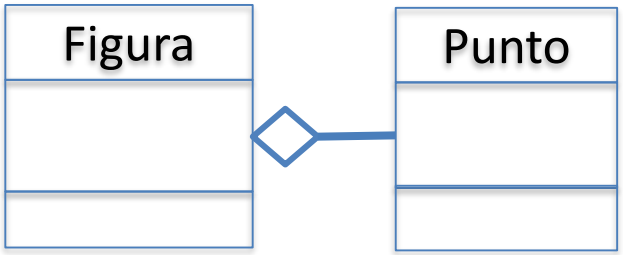
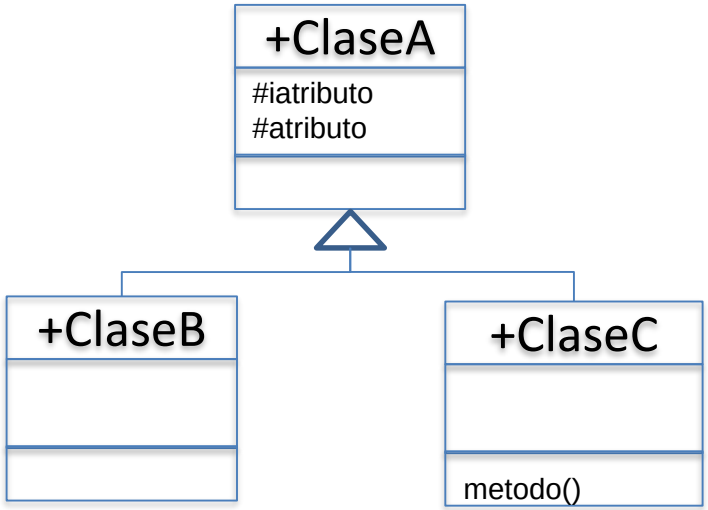
Universidad Distrital Francisco José de caldas
Tecnología en Sistematización de Datos



Relaciones entre clases
Composición

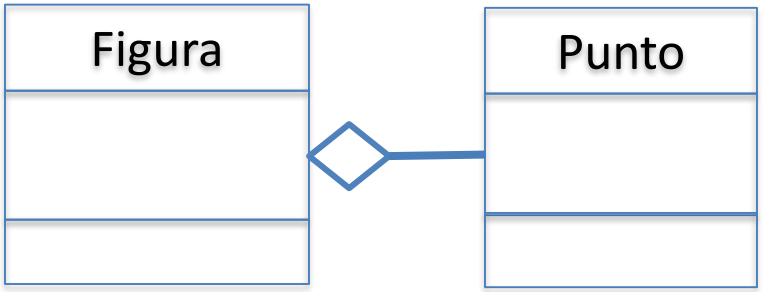


Programación Orientada a Objetos
Sonia Alexandra Pinzón Nuñez
2020

Agregación y Composición

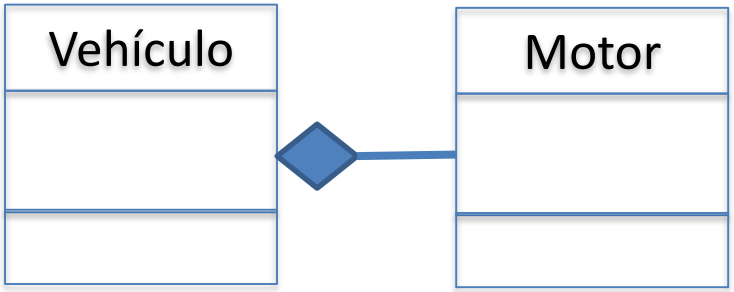
Asociaciones o Relaciones entre un todo y sus partes

Agregación



Las partes o agregaciones pueden formar parte de otros objetos o ser independientes

Composición



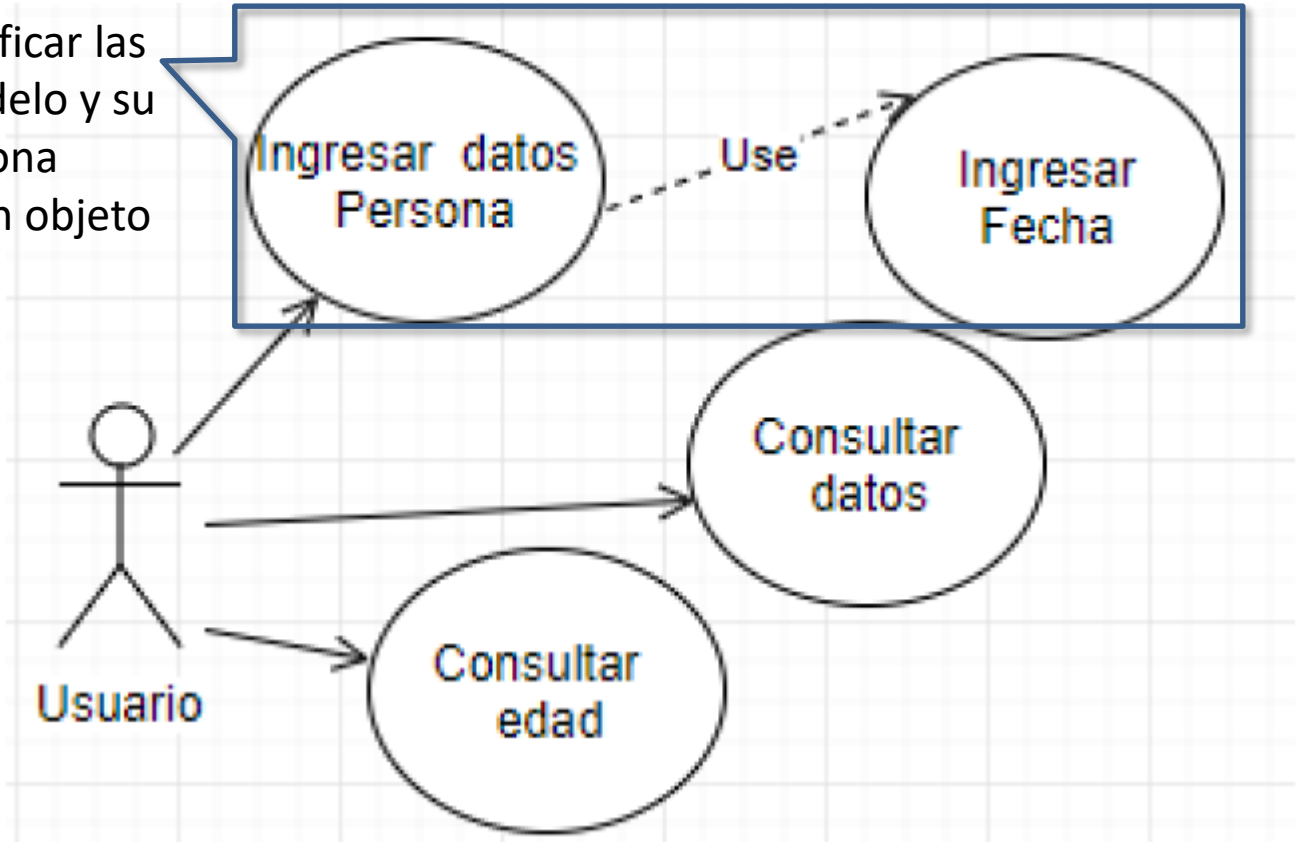
Las partes o componentes solo existen asociadas al objeto.

Agregación

Ejercicio:

Registrar los datos de una persona (Id,Nombre, Fecha de Nacimiento) y Generar la edad

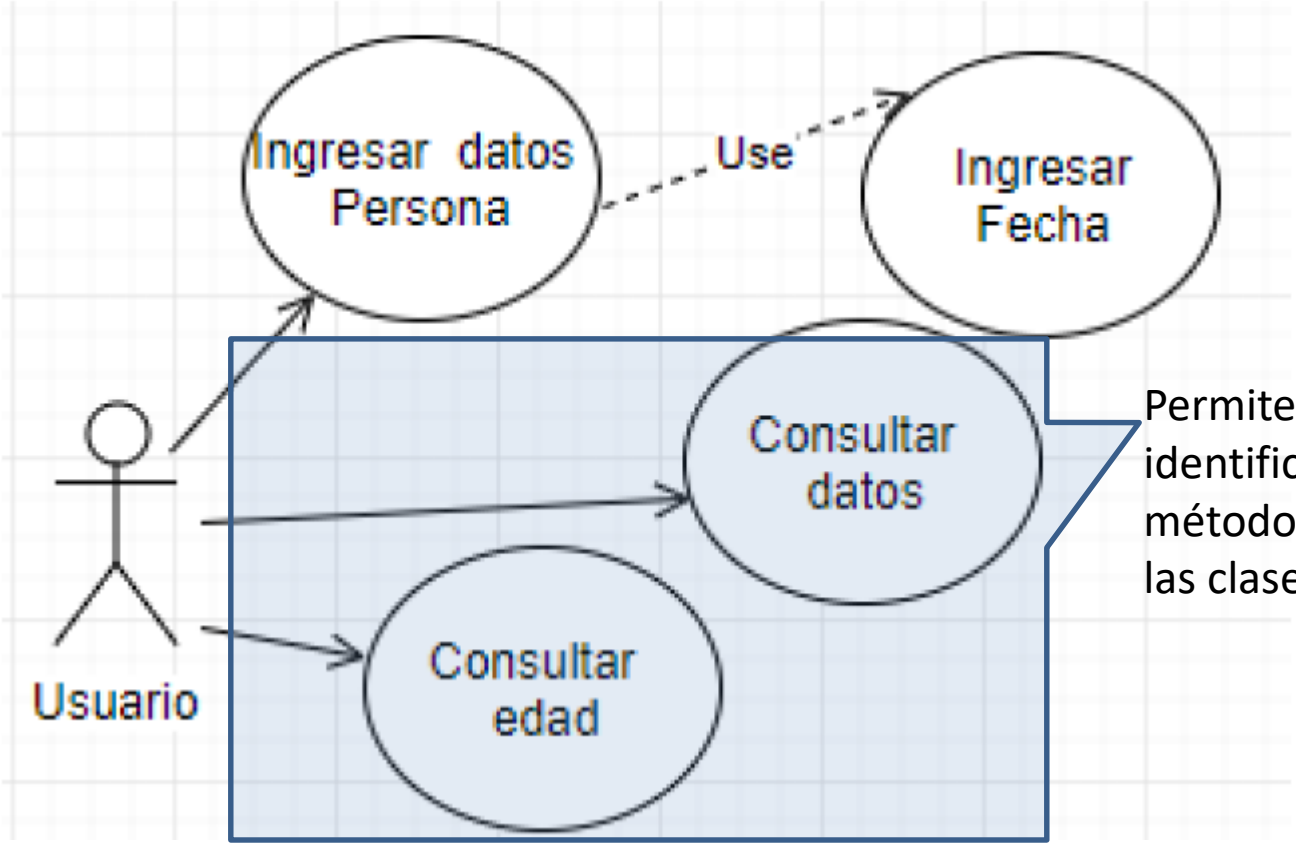
Permite identificar las clases del modelo y su relación (Persona **agrega(usa)** un objeto de Fecha)



Agregación

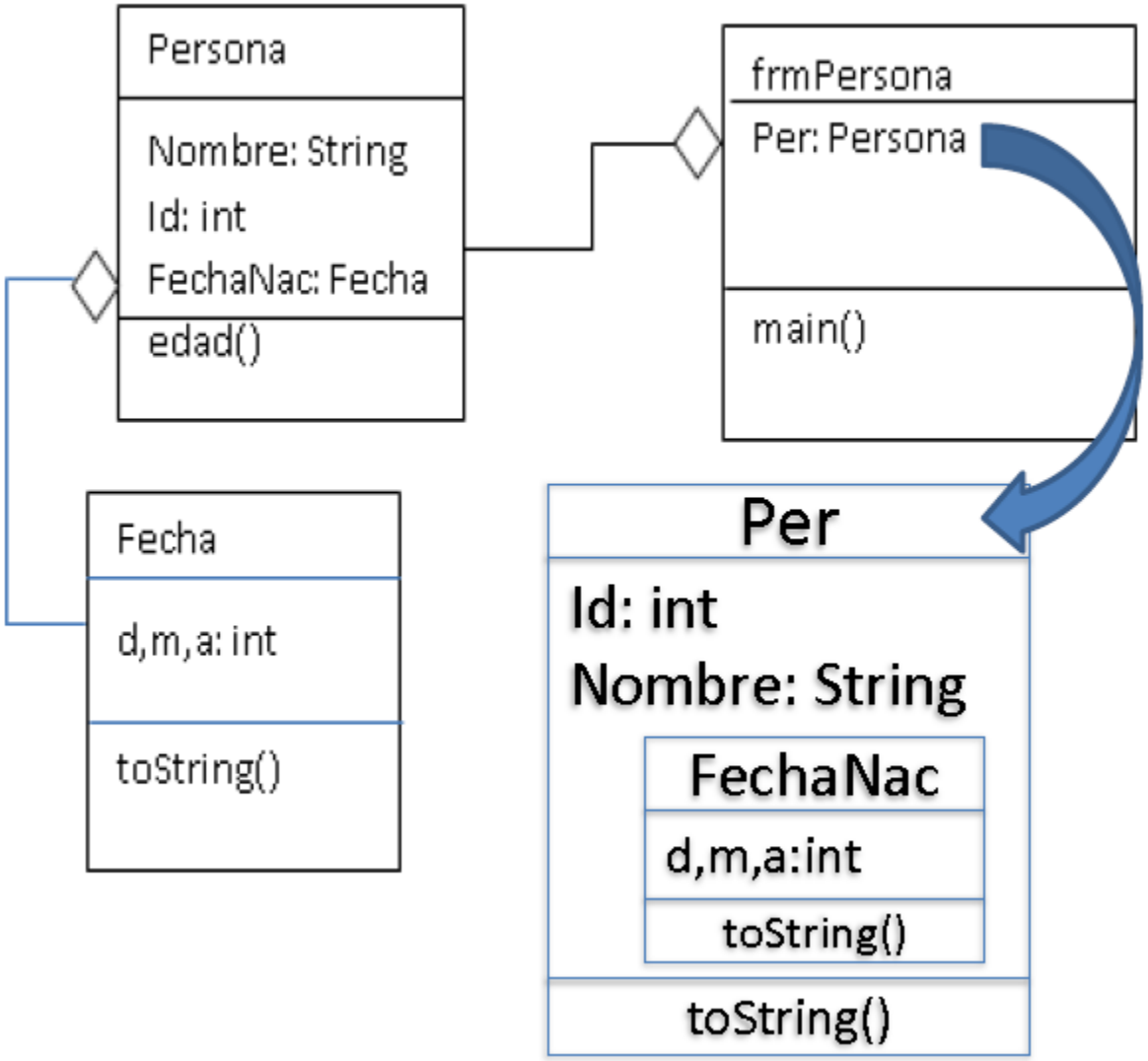
Ejercicio:

Registrar los datos de una persona (Id,Nombre, Fecha de Nacimiento) y Generar la edad

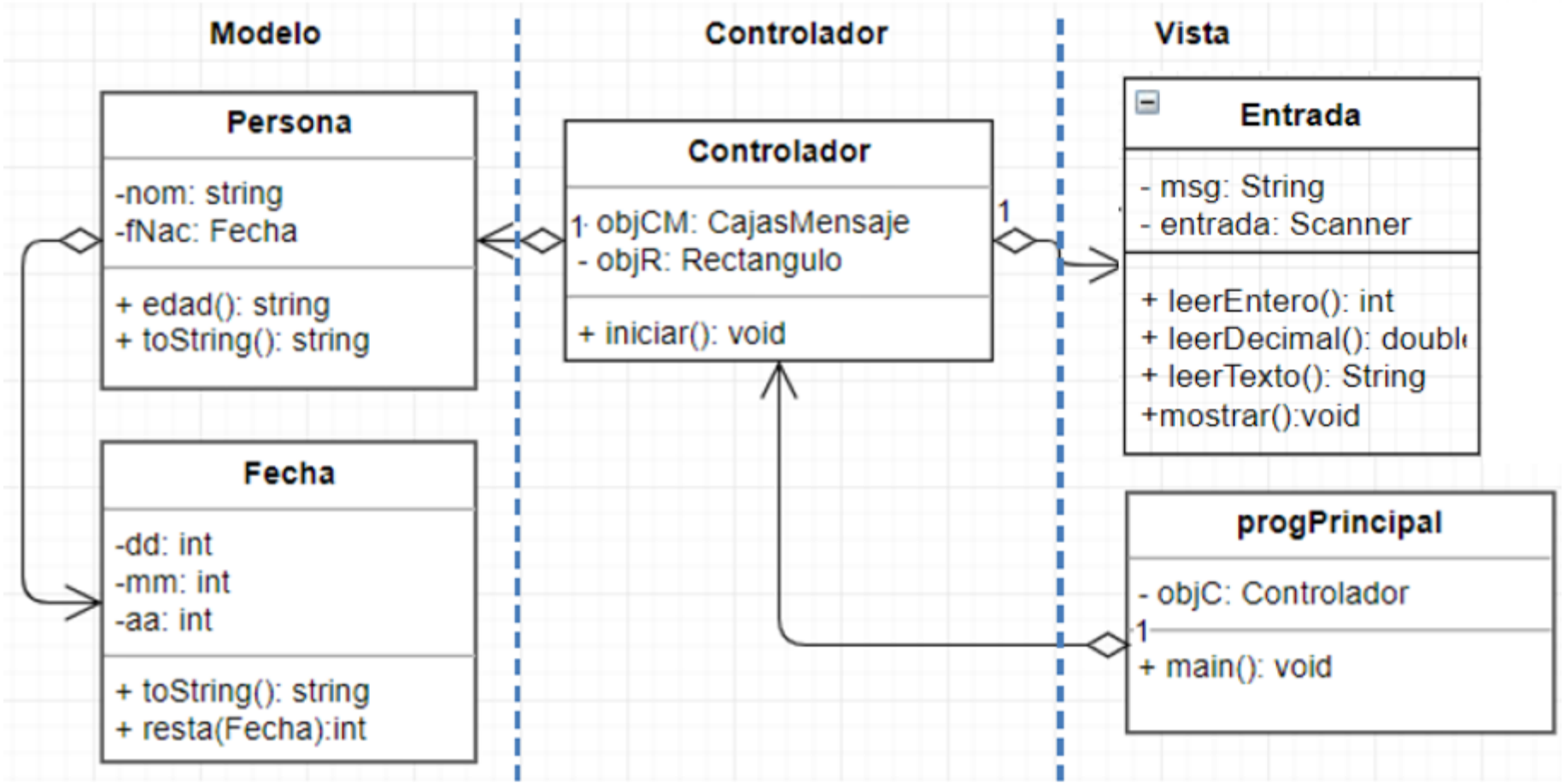


Agregación

Relación en la cual una clase está compuesta o contiene como atributo un objeto de otra clase, por ejemplo, en la figura se puede observar que la capa Lógica contiene una clase denominada Persona cuyos atributos son Nombre, Id y FechaNac (Fecha de Nacimiento).



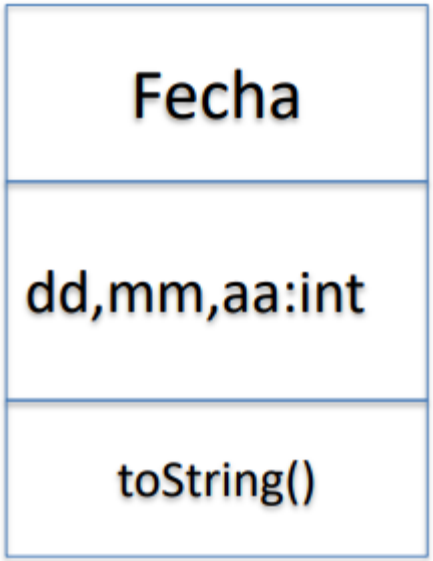
Agregación


Modelo
Controlador
Vista
Persona
-nom: string
-fNac: Fecha
+ edad(): string
+ toString(): string
Fecha
-dd: int
-mm: int
-aa: int
+ toString(): string
+ resta(Fecha):int
Controlador
1
- objCM: CajasMensaje
- objR: Rectangulo
+ iniciar(): void
1
Entrada
- msg: String
- entrada: Scanner
+ leerEntero(): int
+ leerDecimal(): doubl
+ leerTexto(): String
+mostrar():void
progPrincipal
- objC: Controlador
1
+ main(): void

Clase Fecha (Modelo)

```java
public class Fecha {
    private int dd,mm,aa;
    public Fecha(int dd, int mm, int aa) {
        this.dd = dd;
        this.mm = mm;
        this.aa = aa;
    }
    public Fecha() {
        Calendar fechaSis= Calendar.getInstance();
        this.dd = fechaSis.get(Calendar.DAY_OF_MONTH);
        this.mm = fechaSis.get(Calendar.MONTH)+1;//genera datos 0-11
        this.aa = fechaSis.get(Calendar.YEAR);
    }
    public int getDd()
    { return this.dd;
    }
    public void setDd(int d)
    { this.dd=d;
    }
    public int getMm() {
        return mm;
    }
    public void setMm(int mm) {
        this.mm = mm;
    }
    public int getAa() {
        return aa;
    }
    public void setAa(int aa) {
        this.aa = aa;
    }
    @Override
    public String toString()
    {return this.dd+"/"+this.mm+"/"+this.aa;
    }
}
```

La Clase **Calendar** permite obtener la fecha y hora del sistema.

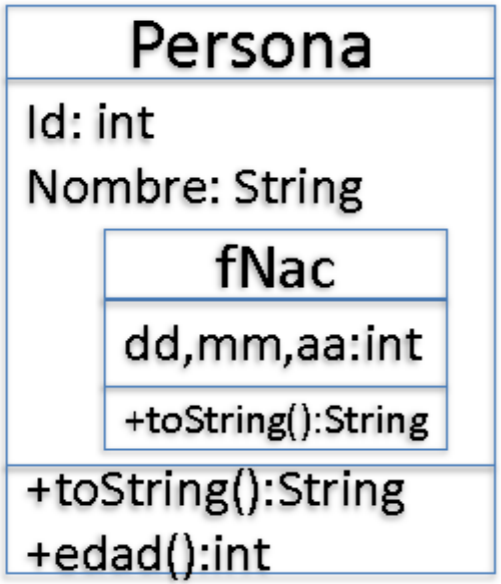


Clase Persona Agregación Fecha (Modelo)

```
public class Persona {
    private String Nom,Id;
    private Fecha fN;
    public Persona(){
     this.Nom="";
     this.Id="";
     this.fN=new Fecha();
    }
    public Persona(String nom, String id, Fecha f){
     this.Nom=nom;
     this.Id=id;
     this.fN=f;
    }
     public String getNom() {...3 lines }
     public void setNom(String Nom) {...3 lines }
     public String getId() {...3 lines }
     public void setId(String Id) {...3 lines }
     public Fecha getfN() {...3 lines }
     public void setfN(Fecha fN) {...3 lines }
     @Override
     public String toString() {
         return "Nombre: " + Nom +
                 "Identificación" + Id +
                 "Fecha Nacimiento " + fN.toString();
     }
    public int edad(){
    Calendar fa=Calendar.getInstance();
    int e;
     e=fa.get(Calendar.YEAR)- fN.getAa();
    return e;
    }
}
```

Definición del atributo fNac que es un objeto de tipo Fecha

El atributo fNac se instancia, es decir, se deberá crear un objeto con valores iniciales usando el constructor básico de la Clase Fecha



Clase Controlador (Control)

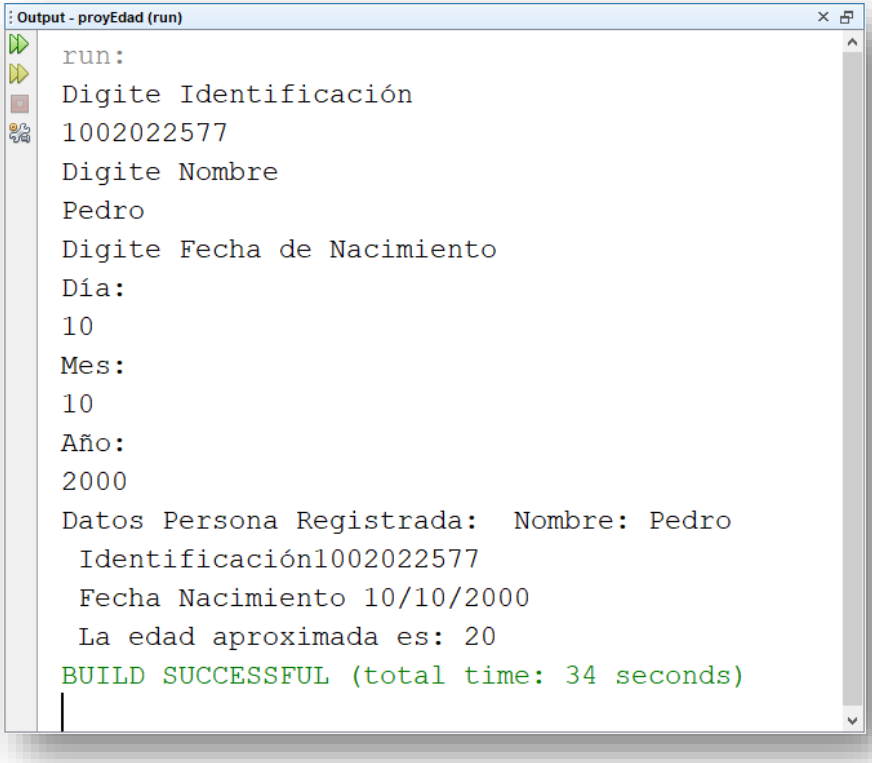
```
public class Controlador {
    Persona objP;
    Entrada objE;
    public Controlador(Persona objP, Entrada objCM) {
        this.objP = objP;
        this.objE = objCM;
    }
      public Controlador() {
        this.objP = new Persona();
        this.objE = new Entrada();
    }
    public void iniciar(){
        objP.setId(objE.leerTexto("Digite Identificación"));
        objP.setNom(objE.leerTexto("Digite Nombre"));
        objE.mostrar("Digite Fecha de Nacimiento ");
        objP.setfN(new Fecha(objE.leerEntero("Día: "),
                             objE.leerEntero("Mes: "),
                             objE.leerEntero("Año: ")));
        objE.mostrar("Datos Persona Registrada: "+ objP.toString()+
                     "\n La edad aproximada es: "+objP.edad());
    }
}
```

Se crea una instancia temporal de tipo Fecha a partir del constructor paramétrico para enviarla al objeto Persona

Programa Autónomo (Proyecto)

```
public class progPrincipal {

    public static void main(String[] args) {
        Controlador objC= new Controlador();
        objC.iniciar();
    }
}
```



Bibliografía

- LADRÓN, Jorge Martínez. Fundamentos de programación en Java - 4ed. Ed. EME.Universidad Complutense de Madrid. Madrid(España), formato Digital
- Deitel y Deitel. Programación Java. Editorial Mc Graw Hill.