

Programación Avanzada, curso 2011
Juan Manuel Fernández Peña

```
//Ejemplo curso Programación Avanzada feb-jun 2011
//Juan Manuel Fernández

public class Persona {
    private String nombre, apellido1, apellido2;
    private int dia, mes, anio;

    // Constructor
    public Persona(String nn, String a1, String a2, int dd, int mm, int
aa){
        nombre = nn; apellido1 = a1; apellido2 = a2;
        dia = dd; mes = mm; anio = aa;
    }

    //métodos
    public String getNombreCompleto(){
        return nombre+" "+apellido1+" "+apellido2;
    }

    public String getNombre(){
        return nombre;
    }

    public String getFechaNacim(){
        return ""+par(dia)+"/"+par(mes)+"/"+anio;
    }

    public int getAnio(){
        return anio;
    }

    public String dameRFC(){
        String rfc = "";
        String aux = ""+anio;
        rfc = rfc+apellido1.substring(0,2);
        rfc += apellido2.substring(0,1);
        rfc += nombre.substring(0,1);
        rfc += aux.substring(2);
        rfc += par(mes);
        rfc += par(dia);
        rfc = rfc.toUpperCase();
        return rfc;
    }

    private String par(int x){
        String resp="";
        resp = resp + x;
        if (resp.length()<2)
            resp = "0"+resp;
        return resp;
    }
}
```

Programación Avanzada, curso 2011
Juan Manuel Fernández Peña

```
/*
 * Ejemplo curso 2011, Juan Manuel Fernández
 * Clase asociada con Persona
 */

public class Dirección {
    private String calle;
    private int numext;
    private String letraext;
    private String numint;
    private String colonia;
    private long codpostal;
    private String ciudad;
    private String estado;

    //Constructor completo
    public Dirección(String cc, int ne, String le, String ni,
        String col, long cp, String cdad, String edo){
        calle = cc; numext = ne; letraext = le; numint = ni;
        colonia = col; codpostal = cp; ciudad = cdad; estado = edo;
    }
    //forma mínima
    public Dirección(String cc, int ne,
        String col, long cp, String cdad, String edo){
        calle = cc; numext = ne; letraext = " "; numint = " ";
        colonia = col; codpostal = cp; ciudad = cdad; estado = edo;
    }

    public String getCalle() {
        return calle;
    }

    public int getNumext() {
        return numext;
    }

    public String getLetraext() {
        return letraext;
    }

    public String getNumint() {
        return numint;
    }

    public String getColonia() {
        return colonia;
    }

    public long getCodpostal() {
        return codpostal;
    }

    public String getCiudad() {
        return ciudad;
    }
}
```

Programación Avanzada, curso 2011
Juan Manuel Fernández Peña

```
public String getEstado() {  
    return estado;  
}  
public String getDirecciónCompleta(){  
    String ret=calle+" "+numext;  
    if (!letraext.equals(" "))  
        ret += " "+letraext;  
    if (!numint.equals(" "))  
        ret += " "+numint;  
    ret += "\n";  
    ret += colonia+" CP "+codpostal+"\n";  
    ret += ciudad+", "+estado;  
    return ret;  
}  
}
```

Observar dos constructores con diferente número de parámetros

Observe el uso de caracteres de control en el texto (\n para salto de línea)

Cuando hay muchos get y set, en Eclipse use botón derecho -> Source -> getters y setters y elija los que quiere.

```
/*
 * Ejemplo curso 2011, Juan Manuel Fernández
 * Clase derivada de Persona a la que se agregó su dirección como otra
clase
 */

public class Personad extends Persona {
    Dirección domicilio=null;

    public Personad(String nn, String a1, String a2, int dd, int mm,
int aa){
        super(nn, a1, a2, dd, mm, aa);
    }

    public boolean setDomicilio(Dirección dir){
        if (dir != null)
            domicilio = dir;
        return (domicilio != null);
    }

    public Dirección getDomicilio(){
        return domicilio;
    }
}
```

Observe que la Personad puede tener una dirección, pero esta se asigna a través de un método, no nace con ella, es decir, la dirección es un objeto independiente, que puede ser compartido por varios objetos de tipo Personad.

Programación Avanzada, curso 2011
Juan Manuel Fernández Peña

```
import junit.framework.TestCase;
/*
 * Ejemplo curso 2011, Juan Manuel Fernández
 * Clase Para probar Personad
 */

public class PersonadTest extends TestCase {

    Personad amigo; Dirección dirDeAmigo;

    protected void setUp() throws Exception {
        super.setUp();
        amigo = new Personad("Ifigenia", "Zamudio", "Álvarez", 13, 6,
2000);
        dirDeAmigo = new Dirección("Alfalfa", 27, "Avanzada", 65021,
"San Nicolás de las Tunas", "Hidalgo");
    }

    public void testSetDomicilio() {
        amigo.setDomicilio(dirDeAmigo);
        System.out.println("Nombre: "+amigo.getNombreCompleto()+
" Dirección:"+"\n"
+amigo.getDomicilio().getDirecciónCompleta());
        assertEquals("Alfalfa", amigo.getDomicilio().getCalle());
    }

}
}
```

Observe el uso de métodos de objetos asociados con otro objeto, usando delegación. Note que la dirección no es obligatoria y se asigna a través de un método, en momento distinto a la creación de los objetos.