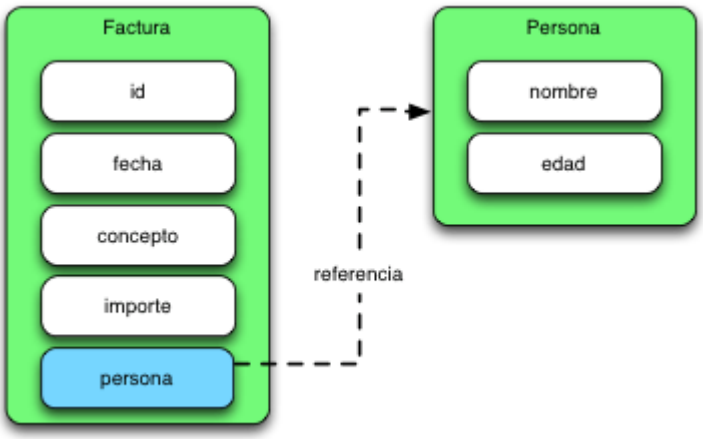
Hemos visto en post anteriores como trabajar con JPA a nivel de clases pero no hemos relacionado clases entre si .Vamos a cubrir en este articulo ese punto para ello vamos a construir dos clases de persistencia (Factura y Persona) entre las cuales existe una relación de 1 a n . Una Persona por lo tanto tiene n facturas y una Factura pertenece a una persona . El concepto de Factura es el siguiente.

Mientras que el concepto de Persona es mucho mas sencillo de entrada ya que solo dispone de nombre y edad.

En estos momentos por su definición ambas entidades son independientes . Es momento de relacionarlas para ello vamos a comenzar con la clase Factura. Toda Factura pertenece a una Persona en concreto y por lo tanto deberemos crear un campo de tipo Persona en la clase Factura.

Al crear una referencia estamos construyendo una relación entre ambos conceptos.



Vamos a verlo en código :

```
<br>
package com.arquitecturajava;</p>
<p>import java.util.Date;</p>
<p>public class Factura {<br>
        private int id;</p>
<p>       private Date fecha;<br>
        private double importe;<br>
        private String concepto;<br>
        private Persona persona;</p>
<p>       public Factura() {<br>
              super();</p>
<p>       }<br>
        public Factura( Date fecha, double importe, String
concepto,Persona persona) {<br>
              super();</p>
<p>               this.fecha = fecha;<br>
              this.importe = importe;<br>
              this.concepto = concepto;<br>
              this.persona=persona;<br>
        }</p>
<p>       public Persona getPersona() {<br>
              return persona;<br>
        }</p>
<p>       public void setPersona(Persona persona) {<br>
              this.persona = persona;<br>
        }</p>
<p>       // métodos set y get restantes<br>
}
```

Por ahora lo único que hemos creado es una referencia a la clase Persona sin utilizar JPA para nada .El siguiente paso será anotar la clase con anotaciones de JPA para que se construya la relación a nivel de persistencia.

```
</p>
<p>package es.curso.bo;</p>
<p>import java.util.Date;</p>
<p>import javax.persistence.Entity;<br>
import javax.persistence.Id;<br>
import javax.persistence.JoinColumn;<br>
import javax.persistence.ManyToOne;</p>
<p>@Entity<br>
public class Factura {</p>
<p>       @Id<br>
        private int id;<br>
        private Date fecha;<br>
        private double importe;<br>
        private String concepto;</p>
<p>       @ManyToOne<br>
        @JoinColumn(name="persona_nombre")<br>
        private Persona persona;</p>
<p>       public Factura() {<br>
              super();<br>
              // TODO Auto-generated constructor stub<br>
        }</p>
<p>       public Factura(int id, Date fecha, double importe, String
concepto,Persona persona) {<br>
              super();<br>
              this.id=id;<br>
              this.fecha = fecha;<br>
              this.importe = importe;<br>
              this.concepto = concepto;<br>
              this.persona=persona;<br>
        }</p>
<p>       public Persona getPersona() {<br>
```

```
                return persona;<br>
        }</p>
<p>        public void setPersona(Persona persona) {<br>
                this.persona = persona;<br>
        }<br>
```

@ManyToOne :Es una de las anotaciones mas habituales a nivel de JPA y se encarga de generar una relación de muchos a uno .
@JoinColumn:Esta anotación sirve en JPA para hacer referencia a la columna que es clave externa en la tabla y que se encarga de definir la relación . En nuestro caso la tabla Factura contendrá una columna persona_nombre con el nombre de la persona propietaria de la factura.

Una vez realizadas estas operaciones ya tenemos la primera parte de la relación construida.Sin embargo aunque puede ser que para algunas aplicaciones esto sea suficiente el framework Hibernate entre sus buenas prácticas recomienda que las relaciones sean bidireccionales. Es decir que tambíen la clase Persona tenga relación con las facturas. En el siguiente Post cubriremos la otra parte.