hibernate Annotation一对一单向外键关联

作者：terry
原文链接：https://ld246.com/article/1453641543788
来源网站：链滴
许可协议：署名-相同方式共享 4.0 国际 (CC BY-SA 4.0)

使用注解的方式进行一对一单向外键关联设置。如下：
（1）Wife类

<ol class="dp-j" start="1">
<li class="alt">package com.yy.hiber
ate.model;

<li class="alt">import javax.persisten
e.Entity;
import javax.persistence.Generat
dValue;
<li class="alt">import javax.persisten
e.Id;

<li class="alt">@Entity </s
an>
public <span class="ke
word">class Wife {
<li class="alt">
   private&n
sp;int id; </sp
n>
<li class="alt">   private
span> String name;

<li class="alt">   @Id
span>
   @GeneratedValue</spa
>
<li class="alt">   public<
pan> int getId() {

   <span class="keyword
>return id;
<li class="alt">   }
   public&nb
p;void setId(<span class="k
yword">int id) {
<li class="alt">   <span class
"keyword">this.id = id;
   }
<li class="alt">   public<
pan> String getName() {
   <span class="keyword
>return name;
<li class="alt">   }
   public&nb
p;void setName(String name)
nbsp;{
<li class="alt">   <span class
"keyword">this.name = name; <
li>
   }

<li class="alt">}

（2）Husband类

<ol class="dp-j" start="1">
<li class="alt">package com.yy.hiber
ate.model;

<li class="alt">import javax.persisten
e.Entity;
import javax.persistence.Generat
dValue;
<li class="alt">import javax.persisten
e.Id;
import javax.persistence.JoinCol
mn;
<li class="alt">import javax.persisten
e.OneToOne;

<li class="alt">@Entity </s
an>
public <span class="ke
word">class Husband {
<li class="alt">
   private&n
sp;int id; </sp
n>
<li class="alt">   private
span> String name;

<li class="alt">   private
span> Wife wife;

<li class="alt">   @Id
span>
   @GeneratedValue</spa
>
<li class="alt">   public<
pan> int getId() {

   <span class="keyword
>return id;
<li class="alt">   }
   public&nb
p;String getName() {
<li class="alt">   <span class
"keyword">return name;
   }
<li class="alt">
   @OneToOne<s
an>
<li class="alt">   @JoinColum
(name="wifeId") &nbs

;
   public&nbp;Wife getWife() {
<li class="alt">   <span class"keyword">return wife;
   }
<li class="alt">   public<pan> void setId(<spn class="keyword">int id) {
   <span class="keyword>this.id = id;
<li class="alt">   }
   public&nbp;void setName(String name)nbsp;{
<li class="alt">   <span class"keyword">this.name = name;
   }
<li class="alt">   public<pan> void setWife(Wife wife) {
   <span class="keyword>this.wife = wife;
<li class="alt">   }
}

（3）SchemaExport

<ol class="dp-j" start="1">
<li class="alt">new SchemaExport(</pan>new AnnotationConfiguration().configur()).create(false, <span class"keyword">true);

自动生成的表如下：

<ol class="dp-xml" start="1">
<li class="alt">mysql> desc husband;
+--------+--------------+------+-----+---------+----------------+ </san>
<li class="alt">| Field | Type   nbsp; | Null | Key | Default | Extra | nbsp;
+--------+--------------+------+-----+---------+----------------+ </san>
<li class="alt">| id | int(11) &bsp; | NO | PRI | ULL | auto_increment |

| name | varchar(255) | YES
 | | NULL |

  |
<li class="alt">| wifeId | int(11)
 | YES | MUL | NULL   &nbs
;| &nbs
; |
+--------+--------------+------+-----+---------+----------------+ </s
an>

<ol class="dp-xml" start="1">
<li class="alt">mysql>
desc wife;
+-------+--------------+------+-----+---------+----------------+ </sp
n>
<li class="alt">| Field | Type
nbsp; | Null | Key | Default |&nbs
;Extra |
/span>
+-------+--------------+------+-----+---------+----------------+ </sp
n>
<li class="alt">| id | int(11) &
bsp; | NO | PRI | NULL&n
sp; | auto_increment |
| name | varchar(255) | YES
     | NULL |

 |
<li class="alt">+-------+--------------+------+-----+---------+----------------+
nbsp;