### 一个简单的例子.创建数据库对象

要求：表、序列、索引建在应用数据库用户下，表放在数据表空间中，索引放在索引表空间中；视图、包建在APPS下，表和序列需要在APPS下创建别名。

本小节的脚本在PL/SQL Developer中，用APPS登录，在Command Window中运行。

#### 创建数据库对象

这里仅创建“头信息”表，并遵循如下规范：

1. 1个表关键字ID，通常与表名一致，并用Sequence为每条记录获得一个唯一值。
2. 1个组织ID，根据不同的开发选用不同层次的组织ID，这里的销售订单跑在OU层，所以基表命名为\_ALL表，并创建一个过滤组织的View作为“基表”。
3. 5个Who字段，记录由谁在何时创建，并由谁在何时修改，登录ID是多少。
4. 4个请求字段，处理的请求ID、请求日期、并发程序ID及其应用ID，如果有的话。
5. 16个描述性弹性域字段，1个上下文字段，15个弹性域字段。
6. 另外，根据以往开发经验，加个描述字段可解决不少难缠的问题。
7. 因为脚本中带有Schema前缀，可以在APPS直接运行创建，源文件为cux\_order\_headers\_all.sql。

*-- Create table*  
**create** **table** SCF.CUX\_ORDER\_HEADERS\_ALL  
(  
 HEADER\_ID **NUMBER** **not** **null**,  
 ORG\_ID **NUMBER** **not** **null**,  
 ORDER\_NUMBER **NUMBER** **not** **null**,  
 ORDERED\_DATE **DATE** **not** **null**,  
 ORDER\_TYPE **VARCHAR2**(1) **not** **null**,  
 CUSTOMER\_ID **NUMBER** **not** **null**,  
 SALESREP\_ID **NUMBER**(15),  
 PRICE\_LIST\_ID **NUMBER**,  
 CURRENCY\_CODE **VARCHAR2**(15) **not** **null**,  
 FLOW\_STATUS\_CODE **VARCHAR2**(30) **not** **null**,  
 **DESCRIPTION** **VARCHAR2**(240),  
 CREATION\_DATE **DATE** **not** **null**,  
 CREATED\_BY **NUMBER** **not** **null**,  
 LAST\_UPDATED\_BY **NUMBER** **not** **null**,  
 LAST\_UPDATE\_DATE **DATE** **not** **null**,  
 LAST\_UPDATE\_LOGIN **NUMBER**,  
 PROGRAM\_APPLICATION\_ID **NUMBER**,  
 PROGRAM\_ID **NUMBER**,  
 PROGRAM\_UPDATE\_DATE **DATE**,  
 REQUEST\_ID **NUMBER**,  
 ATTRIBUTE\_CATEGORY **VARCHAR2**(30),  
 ATTRIBUTE1 **VARCHAR2**(240),  
 ATTRIBUTE2 **VARCHAR2**(240),  
 ATTRIBUTE3 **VARCHAR2**(240),  
 ATTRIBUTE4 **VARCHAR2**(240),  
 ATTRIBUTE5 **VARCHAR2**(240),  
 ATTRIBUTE6 **VARCHAR2**(240),  
 ATTRIBUTE7 **VARCHAR2**(240),  
 ATTRIBUTE8 **VARCHAR2**(240),  
 ATTRIBUTE9 **VARCHAR2**(240),  
 ATTRIBUTE10 **VARCHAR2**(240),  
 ATTRIBUTE11 **VARCHAR2**(240),  
 ATTRIBUTE12 **VARCHAR2**(240),  
 ATTRIBUTE13 **VARCHAR2**(240),  
 ATTRIBUTE14 **VARCHAR2**(240),  
 ATTRIBUTE15 **VARCHAR2**(240)  
)  
**tablespace** SCF;  
  
*-- Create/Recreate indexes*   
**create** **unique** **index** SCF.CUX\_ORDER\_HEADERS\_U1 **on** SCF.CUX\_ORDER\_HEADERS\_ALL (HEADER\_ID)  
 **tablespace** SCF;  
   
**create** **unique** **index** SCF.CUX\_ORDER\_HEADERS\_U2 **on** SCF.CUX\_ORDER\_HEADERS\_ALL (ORG\_ID, ORDER\_NUMBER)  
 **tablespace** SCF;  
  
**create** **index** SCF.CUX\_ORDER\_HEADERS\_N1 **on** SCF.CUX\_ORDER\_HEADERS\_ALL (CUSTOMER\_ID)  
 **tablespace** SCF;  
   
**create** **index** SCF.CUX\_ORDER\_HEADERS\_N2 **on** SCF.CUX\_ORDER\_HEADERS\_ALL (SALESREP\_ID)  
 **tablespace** SCF;  
   
**create** **index** SCF.CUX\_ORDER\_HEADERS\_N3 **on** SCF.CUX\_ORDER\_HEADERS\_ALL (FLOW\_STATUS\_CODE)  
 **tablespace** SCF;  
   
*-- Create/Recreate sequence*  
**CREATE** **SEQUENCE** SCF.CUX\_ORDER\_HEADERS\_S;  
  
*-- Create/Recreate synonum*  
**CREATE** **SYNONYM** CUX\_ORDER\_HEADERS\_S **FOR** scf.CUX\_ORDER\_HEADERS\_S;  
  
**CREATE** **SYNONYM** CUX\_ORDER\_HEADERS\_ALL **FOR** scf.CUX\_ORDER\_HEADERS\_ALL;  
  
*-- Create/Recreate View*  
**CREATE** **OR** **REPLACE** **VIEW** CUX\_ORDER\_HEADERS **AS**  
**SELECT** header\_id,  
 org\_id,  
 order\_number,  
 ordered\_date,  
 order\_type,  
 customer\_id,  
 salesrep\_id,  
 price\_list\_id,  
 currency\_code,  
 flow\_status\_code,  
 **description**,  
 creation\_date,  
 created\_by,  
 last\_updated\_by,  
 last\_update\_date,  
 last\_update\_login,  
 program\_application\_id,  
 program\_id,  
 program\_update\_date,  
 request\_id,  
 attribute\_category,  
 attribute1,  
 attribute2,  
 attribute3,  
 attribute4,  
 attribute5,  
 attribute6,  
 attribute7,  
 attribute8,  
 attribute9,  
 attribute10,  
 attribute11,  
 attribute12,  
 attribute13,  
 attribute14,  
 attribute15  
 **FROM** cux\_order\_headers\_all  
 **WHERE** nvl(org\_id,  
 nvl(to\_number(decode(substrb(userenv('CLIENT\_INFO'), 1, 1),  
 ' ',  
 **NULL**,  
 substrb(userenv('CLIENT\_INFO'), 1, 10))),  
 -99)) =  
 nvl(to\_number(decode(substrb(userenv('CLIENT\_INFO'), 1, 1),  
 ' ',  
 **NULL**,  
 substrb(userenv('CLIENT\_INFO'), 1, 10))),  
 -99);

#### 注册表和字段

需要向EBS注册表和字段，这样以后就可通过标准功能设置弹性域、监控表操作。

**EXECUTE** AD\_DD.REGISTER\_TABLE('SCF','CUX\_ORDER\_HEADERS\_ALL','T',2,10,40);  
**EXECUTE** AD\_DD.REGISTER\_COLUMN('SCF','CUX\_ORDER\_HEADERS\_ALL','HEADER\_ID',1,'NUMBER',38,'N','N');  
**EXECUTE** AD\_DD.REGISTER\_COLUMN('SCF','CUX\_ORDER\_HEADERS\_ALL','ORG\_ID',2,'NUMBER',38,'N','N');  
**EXECUTE** AD\_DD.REGISTER\_COLUMN('SCF','CUX\_ORDER\_HEADERS\_ALL','ORDER\_NUMBER',3,'NUMBER',38,'N','N');  
**EXECUTE** AD\_DD.REGISTER\_COLUMN('SCF','CUX\_ORDER\_HEADERS\_ALL','ORDERED\_DATE',4,'DATE',9,'N','N');  
**EXECUTE** AD\_DD.REGISTER\_COLUMN('SCF','CUX\_ORDER\_HEADERS\_ALL','ORDER\_TYPE',5,'VARCHAR2',1,'N','N');  
**EXECUTE** AD\_DD.REGISTER\_COLUMN('SCF','CUX\_ORDER\_HEADERS\_ALL','CUSTOMER\_ID',6,'NUMBER',38,'N','N');  
**EXECUTE** AD\_DD.REGISTER\_COLUMN('SCF','CUX\_ORDER\_HEADERS\_ALL','SALESREP\_ID',7,'NUMBER',38,'Y','N');  
**EXECUTE** AD\_DD.REGISTER\_COLUMN('SCF','CUX\_ORDER\_HEADERS\_ALL','PRICE\_LIST\_ID',8,'NUMBER',38,'Y','N');  
**EXECUTE** AD\_DD.REGISTER\_COLUMN('SCF','CUX\_ORDER\_HEADERS\_ALL','CURRENCY\_CODE',9,'VARCHAR2',15,'N','N');  
**EXECUTE** AD\_DD.REGISTER\_COLUMN('SCF','CUX\_ORDER\_HEADERS\_ALL','FLOW\_STATUS\_CODE',10,'VARCHAR2',30,'N','N');  
**EXECUTE** AD\_DD.REGISTER\_COLUMN('SCF','CUX\_ORDER\_HEADERS\_ALL','DESCRIPTION',11,'VARCHAR2',240,'Y','N');  
**EXECUTE** AD\_DD.REGISTER\_COLUMN('SCF','CUX\_ORDER\_HEADERS\_ALL','CREATION\_DATE',12,'DATE',9,'N','N');  
**EXECUTE** AD\_DD.REGISTER\_COLUMN('SCF','CUX\_ORDER\_HEADERS\_ALL','CREATED\_BY',13,'NUMBER',38,'N','N');  
**EXECUTE** AD\_DD.REGISTER\_COLUMN('SCF','CUX\_ORDER\_HEADERS\_ALL','LAST\_UPDATED\_BY',14,'NUMBER',38,'N','N');  
**EXECUTE** AD\_DD.REGISTER\_COLUMN('SCF','CUX\_ORDER\_HEADERS\_ALL','LAST\_UPDATE\_DATE',15,'DATE',9,'N','N');  
**EXECUTE** AD\_DD.REGISTER\_COLUMN('SCF','CUX\_ORDER\_HEADERS\_ALL','LAST\_UPDATE\_LOGIN',16,'NUMBER',38,'Y','N');  
**EXECUTE** AD\_DD.REGISTER\_COLUMN('SCF','CUX\_ORDER\_HEADERS\_ALL','PROGRAM\_APPLICATION\_ID',17,'NUMBER',38,'Y','N');  
**EXECUTE** AD\_DD.REGISTER\_COLUMN('SCF','CUX\_ORDER\_HEADERS\_ALL','PROGRAM\_ID',18,'NUMBER',38,'Y','N');  
**EXECUTE** AD\_DD.REGISTER\_COLUMN('SCF','CUX\_ORDER\_HEADERS\_ALL','PROGRAM\_UPDATE\_DATE',19,'DATE',9,'Y','N');  
**EXECUTE** AD\_DD.REGISTER\_COLUMN('SCF','CUX\_ORDER\_HEADERS\_ALL','REQUEST\_ID',20,'NUMBER',38,'Y','N');  
**EXECUTE** AD\_DD.REGISTER\_COLUMN('SCF','CUX\_ORDER\_HEADERS\_ALL','ATTRIBUTE\_CATEGORY',21,'VARCHAR2',30,'Y','N');  
**EXECUTE** AD\_DD.REGISTER\_COLUMN('SCF','CUX\_ORDER\_HEADERS\_ALL','ATTRIBUTE1',22,'VARCHAR2',240,'Y','N');  
**EXECUTE** AD\_DD.REGISTER\_COLUMN('SCF','CUX\_ORDER\_HEADERS\_ALL','ATTRIBUTE2',23,'VARCHAR2',240,'Y','N');  
**EXECUTE** AD\_DD.REGISTER\_COLUMN('SCF','CUX\_ORDER\_HEADERS\_ALL','ATTRIBUTE3',24,'VARCHAR2',240,'Y','N');  
**EXECUTE** AD\_DD.REGISTER\_COLUMN('SCF','CUX\_ORDER\_HEADERS\_ALL','ATTRIBUTE4',25,'VARCHAR2',240,'Y','N');  
**EXECUTE** AD\_DD.REGISTER\_COLUMN('SCF','CUX\_ORDER\_HEADERS\_ALL','ATTRIBUTE5',26,'VARCHAR2',240,'Y','N');  
**EXECUTE** AD\_DD.REGISTER\_COLUMN('SCF','CUX\_ORDER\_HEADERS\_ALL','ATTRIBUTE6',27,'VARCHAR2',240,'Y','N');  
**EXECUTE** AD\_DD.REGISTER\_COLUMN('SCF','CUX\_ORDER\_HEADERS\_ALL','ATTRIBUTE7',28,'VARCHAR2',240,'Y','N');  
**EXECUTE** AD\_DD.REGISTER\_COLUMN('SCF','CUX\_ORDER\_HEADERS\_ALL','ATTRIBUTE8',29,'VARCHAR2',240,'Y','N');  
**EXECUTE** AD\_DD.REGISTER\_COLUMN('SCF','CUX\_ORDER\_HEADERS\_ALL','ATTRIBUTE9',30,'VARCHAR2',240,'Y','N');  
**EXECUTE** AD\_DD.REGISTER\_COLUMN('SCF','CUX\_ORDER\_HEADERS\_ALL','ATTRIBUTE10',31,'VARCHAR2',240,'Y','N');  
**EXECUTE** AD\_DD.REGISTER\_COLUMN('SCF','CUX\_ORDER\_HEADERS\_ALL','ATTRIBUTE11',32,'VARCHAR2',240,'Y','N');  
**EXECUTE** AD\_DD.REGISTER\_COLUMN('SCF','CUX\_ORDER\_HEADERS\_ALL','ATTRIBUTE12',33,'VARCHAR2',240,'Y','N');  
**EXECUTE** AD\_DD.REGISTER\_COLUMN('SCF','CUX\_ORDER\_HEADERS\_ALL','ATTRIBUTE13',34,'VARCHAR2',240,'Y','N');  
**EXECUTE** AD\_DD.REGISTER\_COLUMN('SCF','CUX\_ORDER\_HEADERS\_ALL','ATTRIBUTE14',35,'VARCHAR2',240,'Y','N');  
**EXECUTE** AD\_DD.REGISTER\_COLUMN('SCF','CUX\_ORDER\_HEADERS\_ALL','ATTRIBUTE15',36,'VARCHAR2',240,'Y','N');

#### 创建用于Form的View

Form可以直接基于基表，但对于复杂的表，则必需使用View，将各个ID转换为有意义的编码或者描述。

**CREATE** **OR** **REPLACE** **VIEW** CUX\_ORDER\_HEADERS\_V **AS**  
**SELECT** coh.**ROWID** row\_id,  
 coh.header\_id,  
 coh.org\_id,  
 coh.order\_number,  
 coh.ordered\_date,  
 coh.order\_type,  
 coh.customer\_id,  
 rac.customer\_name,  
 coh.salesrep\_id,  
 jrs.**NAME** salesrep\_name,  
 coh.price\_list\_id,  
 qlh.**NAME** price\_list\_name,  
 coh.currency\_code,  
 coh.flow\_status\_code,  
 coh.**description**,  
 coh.creation\_date,  
 coh.created\_by,  
 coh.last\_updated\_by,  
 coh.last\_update\_date,  
 coh.last\_update\_login,  
 coh.program\_application\_id,  
 coh.program\_id,  
 coh.program\_update\_date,  
 coh.request\_id,  
 coh.attribute\_category,  
 coh.attribute1,  
 coh.attribute2,  
 coh.attribute3,  
 coh.attribute4,  
 coh.attribute5,  
 coh.attribute6,  
 coh.attribute7,  
 coh.attribute8,  
 coh.attribute9,  
 coh.attribute10,  
 coh.attribute11,  
 coh.attribute12,  
 coh.attribute13,  
 coh.attribute14,  
 coh.attribute15  
 **FROM** cux\_order\_headers coh,  
 ra\_customers rac,  
 jtf\_rs\_salesreps jrs,  
 qp\_list\_headers\_vl qlh  
 **WHERE** coh.customer\_id = rac.customer\_id  
 **AND** coh.salesrep\_id = jrs.salesrep\_id(+)  
 **AND** coh.price\_list\_id = qlh.list\_header\_id(+)

#### 创建表操作API

不管是基于基表还是基于视图的Block，都建议编写ON-UPDATE、ON-INSERT、ON-DELETE、ON-LOCK触发器，并且，把具体的DML和锁记录代码放入数据库Package中，然后在Form中调用，该Package以后还可以在其他地方调用。这些设计，都是基于以往开发经验所归纳出来的，有利于模块化开发、有利于扩展。以下脚本在APPS下运行，源文件为cux\_order\_headers\_pkg.pck：

**CREATE** **OR** **REPLACE** **PACKAGE** cux\_order\_headers\_pkg **AS**  
  
 */\*=====================================  
 \*\* PROCEDURE: insert\_row()  
 \*\*=====================================\*/*  
 **PROCEDURE** insert\_row(x\_row\_id **IN** **OUT** **VARCHAR2**,  
 x\_header\_id **IN** **OUT** **NUMBER**,  
 p\_org\_id **IN** **NUMBER**,  
 p\_order\_number **IN** **NUMBER**,  
 p\_ordered\_date **IN** **DATE**,  
 p\_order\_type **IN** **VARCHAR2**,  
 p\_customer\_id **IN** **NUMBER**,  
 p\_salesrep\_id **IN** **NUMBER** **DEFAULT** **NULL**,  
 p\_price\_list\_id **IN** **NUMBER** **DEFAULT** **NULL**,  
 p\_currency\_code **IN** **VARCHAR2**,  
 p\_flow\_status\_code **IN** **VARCHAR2**,  
 p\_description **IN** **VARCHAR2** **DEFAULT** **NULL**,  
 p\_creation\_date **IN** **DATE**,  
 p\_created\_by **IN** **NUMBER**,  
 p\_last\_updated\_by **IN** **NUMBER**,  
 p\_last\_update\_date **IN** **DATE**,  
 p\_last\_update\_login **IN** **NUMBER** **DEFAULT** **NULL**,  
 p\_program\_application\_id **IN** **NUMBER** **DEFAULT** **NULL**,  
 p\_program\_id **IN** **NUMBER** **DEFAULT** **NULL**,  
 p\_program\_update\_date **IN** **DATE** **DEFAULT** **NULL**,  
 p\_request\_id **IN** **NUMBER** **DEFAULT** **NULL**,  
 p\_attribute\_category **IN** **VARCHAR2** **DEFAULT** **NULL**,  
 p\_attribute1 **IN** **VARCHAR2** **DEFAULT** **NULL**,  
 p\_attribute2 **IN** **VARCHAR2** **DEFAULT** **NULL**,  
 p\_attribute3 **IN** **VARCHAR2** **DEFAULT** **NULL**,  
 p\_attribute4 **IN** **VARCHAR2** **DEFAULT** **NULL**,  
 p\_attribute5 **IN** **VARCHAR2** **DEFAULT** **NULL**,  
 p\_attribute6 **IN** **VARCHAR2** **DEFAULT** **NULL**,  
 p\_attribute7 **IN** **VARCHAR2** **DEFAULT** **NULL**,  
 p\_attribute8 **IN** **VARCHAR2** **DEFAULT** **NULL**,  
 p\_attribute9 **IN** **VARCHAR2** **DEFAULT** **NULL**,  
 p\_attribute10 **IN** **VARCHAR2** **DEFAULT** **NULL**,  
 p\_attribute11 **IN** **VARCHAR2** **DEFAULT** **NULL**,  
 p\_attribute12 **IN** **VARCHAR2** **DEFAULT** **NULL**,  
 p\_attribute13 **IN** **VARCHAR2** **DEFAULT** **NULL**,  
 p\_attribute14 **IN** **VARCHAR2** **DEFAULT** **NULL**,  
 p\_attribute15 **IN** **VARCHAR2** **DEFAULT** **NULL**);  
  
 */\*=====================================  
 \*\* PROCEDURE: lock\_row()  
 \*\*=====================================\*/*  
 **PROCEDURE** lock\_row(p\_header\_id **IN** **NUMBER**,  
 p\_org\_id **IN** **NUMBER**,  
 p\_order\_number **IN** **NUMBER**,  
 p\_ordered\_date **IN** **DATE**,  
 p\_order\_type **IN** **VARCHAR2**,  
 p\_customer\_id **IN** **NUMBER**,  
 p\_salesrep\_id **IN** **NUMBER**,  
 p\_price\_list\_id **IN** **NUMBER**,  
 p\_currency\_code **IN** **VARCHAR2**,  
 p\_flow\_status\_code **IN** **VARCHAR2**,  
 p\_description **IN** **VARCHAR2**,  
 p\_creation\_date **IN** **DATE**,  
 p\_created\_by **IN** **NUMBER**,  
 p\_last\_updated\_by **IN** **NUMBER**,  
 p\_last\_update\_date **IN** **DATE**,  
 p\_last\_update\_login **IN** **NUMBER**,  
 p\_program\_application\_id **IN** **NUMBER**,  
 p\_program\_id **IN** **NUMBER**,  
 p\_program\_update\_date **IN** **DATE**,  
 p\_request\_id **IN** **NUMBER**,  
 p\_attribute\_category **IN** **VARCHAR2**,  
 p\_attribute1 **IN** **VARCHAR2**,  
 p\_attribute2 **IN** **VARCHAR2**,  
 p\_attribute3 **IN** **VARCHAR2**,  
 p\_attribute4 **IN** **VARCHAR2**,  
 p\_attribute5 **IN** **VARCHAR2**,  
 p\_attribute6 **IN** **VARCHAR2**,  
 p\_attribute7 **IN** **VARCHAR2**,  
 p\_attribute8 **IN** **VARCHAR2**,  
 p\_attribute9 **IN** **VARCHAR2**,  
 p\_attribute10 **IN** **VARCHAR2**,  
 p\_attribute11 **IN** **VARCHAR2**,  
 p\_attribute12 **IN** **VARCHAR2**,  
 p\_attribute13 **IN** **VARCHAR2**,  
 p\_attribute14 **IN** **VARCHAR2**,  
 p\_attribute15 **IN** **VARCHAR2**);  
  
 */\*=====================================  
 \*\* PROCEDURE: update\_row()  
 \*\*=====================================\*/*  
 **PROCEDURE** update\_row(p\_header\_id **IN** **NUMBER**,  
 p\_org\_id **IN** **NUMBER**,  
 p\_order\_number **IN** **NUMBER**,  
 p\_ordered\_date **IN** **DATE**,  
 p\_order\_type **IN** **VARCHAR2**,  
 p\_customer\_id **IN** **NUMBER**,  
 p\_salesrep\_id **IN** **NUMBER** **DEFAULT** **NULL**,  
 p\_price\_list\_id **IN** **NUMBER** **DEFAULT** **NULL**,  
 p\_currency\_code **IN** **VARCHAR2**,  
 p\_flow\_status\_code **IN** **VARCHAR2**,  
 p\_description **IN** **VARCHAR2** **DEFAULT** **NULL**,  
 p\_last\_updated\_by **IN** **NUMBER**,  
 p\_last\_update\_date **IN** **DATE**,  
 p\_last\_update\_login **IN** **NUMBER** **DEFAULT** **NULL**,  
 p\_program\_application\_id **IN** **NUMBER** **DEFAULT** **NULL**,  
 p\_program\_id **IN** **NUMBER** **DEFAULT** **NULL**,  
 p\_program\_update\_date **IN** **DATE** **DEFAULT** **NULL**,  
 p\_request\_id **IN** **NUMBER** **DEFAULT** **NULL**,  
 p\_attribute\_category **IN** **VARCHAR2** **DEFAULT** **NULL**,  
 p\_attribute1 **IN** **VARCHAR2** **DEFAULT** **NULL**,  
 p\_attribute2 **IN** **VARCHAR2** **DEFAULT** **NULL**,  
 p\_attribute3 **IN** **VARCHAR2** **DEFAULT** **NULL**,  
 p\_attribute4 **IN** **VARCHAR2** **DEFAULT** **NULL**,  
 p\_attribute5 **IN** **VARCHAR2** **DEFAULT** **NULL**,  
 p\_attribute6 **IN** **VARCHAR2** **DEFAULT** **NULL**,  
 p\_attribute7 **IN** **VARCHAR2** **DEFAULT** **NULL**,  
 p\_attribute8 **IN** **VARCHAR2** **DEFAULT** **NULL**,  
 p\_attribute9 **IN** **VARCHAR2** **DEFAULT** **NULL**,  
 p\_attribute10 **IN** **VARCHAR2** **DEFAULT** **NULL**,  
 p\_attribute11 **IN** **VARCHAR2** **DEFAULT** **NULL**,  
 p\_attribute12 **IN** **VARCHAR2** **DEFAULT** **NULL**,  
 p\_attribute13 **IN** **VARCHAR2** **DEFAULT** **NULL**,  
 p\_attribute14 **IN** **VARCHAR2** **DEFAULT** **NULL**,  
 p\_attribute15 **IN** **VARCHAR2** **DEFAULT** **NULL**);  
  
 */\*=====================================  
 \*\* PROCEDURE: delete\_row()  
 \*\*=====================================\*/*  
 **PROCEDURE** delete\_row(p\_header\_id **IN** **NUMBER**);  
  
**END**;  
/  
  
**CREATE** **OR** **REPLACE** **PACKAGE** **BODY** cux\_order\_headers\_pkg **AS**  
  
 */\*=====================================  
 \*\* PROCEDURE: insert\_row()  
 \*\*=====================================\*/*  
 **PROCEDURE** insert\_row(x\_row\_id **IN** **OUT** **VARCHAR2**,  
 x\_header\_id **IN** **OUT** **NUMBER**,  
 p\_org\_id **IN** **NUMBER**,  
 p\_order\_number **IN** **NUMBER**,  
 p\_ordered\_date **IN** **DATE**,  
 p\_order\_type **IN** **VARCHAR2**,  
 p\_customer\_id **IN** **NUMBER**,  
 p\_salesrep\_id **IN** **NUMBER** **DEFAULT** **NULL**,  
 p\_price\_list\_id **IN** **NUMBER** **DEFAULT** **NULL**,  
 p\_currency\_code **IN** **VARCHAR2**,  
 p\_flow\_status\_code **IN** **VARCHAR2**,  
 p\_description **IN** **VARCHAR2** **DEFAULT** **NULL**,  
 p\_creation\_date **IN** **DATE**,  
 p\_created\_by **IN** **NUMBER**,  
 p\_last\_updated\_by **IN** **NUMBER**,  
 p\_last\_update\_date **IN** **DATE**,  
 p\_last\_update\_login **IN** **NUMBER** **DEFAULT** **NULL**,  
 p\_program\_application\_id **IN** **NUMBER** **DEFAULT** **NULL**,  
 p\_program\_id **IN** **NUMBER** **DEFAULT** **NULL**,  
 p\_program\_update\_date **IN** **DATE** **DEFAULT** **NULL**,  
 p\_request\_id **IN** **NUMBER** **DEFAULT** **NULL**,  
 p\_attribute\_category **IN** **VARCHAR2** **DEFAULT** **NULL**,  
 p\_attribute1 **IN** **VARCHAR2** **DEFAULT** **NULL**,  
 p\_attribute2 **IN** **VARCHAR2** **DEFAULT** **NULL**,  
 p\_attribute3 **IN** **VARCHAR2** **DEFAULT** **NULL**,  
 p\_attribute4 **IN** **VARCHAR2** **DEFAULT** **NULL**,  
 p\_attribute5 **IN** **VARCHAR2** **DEFAULT** **NULL**,  
 p\_attribute6 **IN** **VARCHAR2** **DEFAULT** **NULL**,  
 p\_attribute7 **IN** **VARCHAR2** **DEFAULT** **NULL**,  
 p\_attribute8 **IN** **VARCHAR2** **DEFAULT** **NULL**,  
 p\_attribute9 **IN** **VARCHAR2** **DEFAULT** **NULL**,  
 p\_attribute10 **IN** **VARCHAR2** **DEFAULT** **NULL**,  
 p\_attribute11 **IN** **VARCHAR2** **DEFAULT** **NULL**,  
 p\_attribute12 **IN** **VARCHAR2** **DEFAULT** **NULL**,  
 p\_attribute13 **IN** **VARCHAR2** **DEFAULT** **NULL**,  
 p\_attribute14 **IN** **VARCHAR2** **DEFAULT** **NULL**,  
 p\_attribute15 **IN** **VARCHAR2** **DEFAULT** **NULL**)  
   
 **IS**  
 **CURSOR** c **IS**  
 **SELECT** **ROWID**  
 **FROM** cux\_order\_headers\_all  
 **WHERE** header\_id = x\_header\_id;  
   
 **BEGIN**  
   
 **IF** x\_header\_id **IS** **NULL** **THEN**  
 **SELECT** cux\_order\_headers\_s.**NEXTVAL** **INTO** x\_header\_id **FROM** dual;  
 **END** **IF**;  
   
 **INSERT** **INTO** cux\_order\_headers\_all  
 (header\_id,  
 org\_id,  
 order\_number,  
 ordered\_date,  
 order\_type,  
 customer\_id,  
 salesrep\_id,  
 price\_list\_id,  
 currency\_code,  
 flow\_status\_code,  
 description,  
 creation\_date,  
 created\_by,  
 last\_updated\_by,  
 last\_update\_date,  
 last\_update\_login,  
 program\_application\_id,  
 program\_id,  
 program\_update\_date,  
 request\_id,  
 attribute\_category,  
 attribute1,  
 attribute2,  
 attribute3,  
 attribute4,  
 attribute5,  
 attribute6,  
 attribute7,  
 attribute8,  
 attribute9,  
 attribute10,  
 attribute11,  
 attribute12,  
 attribute13,  
 attribute14,  
 attribute15)  
 **VALUES**  
 (x\_header\_id,  
 p\_org\_id,  
 p\_order\_number,  
 p\_ordered\_date,  
 p\_order\_type,  
 p\_customer\_id,  
 p\_salesrep\_id,  
 p\_price\_list\_id,  
 p\_currency\_code,  
 p\_flow\_status\_code,  
 p\_description,  
 p\_creation\_date,  
 p\_created\_by,  
 p\_last\_updated\_by,  
 p\_last\_update\_date,  
 p\_last\_update\_login,  
 p\_program\_application\_id,  
 p\_program\_id,  
 p\_program\_update\_date,  
 p\_request\_id,  
 p\_attribute\_category,  
 p\_attribute1,  
 p\_attribute2,  
 p\_attribute3,  
 p\_attribute4,  
 p\_attribute5,  
 p\_attribute6,  
 p\_attribute7,  
 p\_attribute8,  
 p\_attribute9,  
 p\_attribute10,  
 p\_attribute11,  
 p\_attribute12,  
 p\_attribute13,  
 p\_attribute14,  
 p\_attribute15);  
   
 **OPEN** c;  
 **FETCH** c  
 **INTO** x\_row\_id;  
 **IF** (c%**NOTFOUND**) **THEN**  
 **CLOSE** c;  
 **RAISE** no\_data\_found;  
 **END** **IF**;  
 **CLOSE** c;  
   
 **END** insert\_row;  
  
 */\*=====================================  
 \*\* PROCEDURE: lock\_row()  
 \*\*=====================================\*/*  
 **PROCEDURE** lock\_row(p\_header\_id **IN** **NUMBER**,  
 p\_org\_id **IN** **NUMBER**,  
 p\_order\_number **IN** **NUMBER**,  
 p\_ordered\_date **IN** **DATE**,  
 p\_order\_type **IN** **VARCHAR2**,  
 p\_customer\_id **IN** **NUMBER**,  
 p\_salesrep\_id **IN** **NUMBER**,  
 p\_price\_list\_id **IN** **NUMBER**,  
 p\_currency\_code **IN** **VARCHAR2**,  
 p\_flow\_status\_code **IN** **VARCHAR2**,  
 p\_description **IN** **VARCHAR2**,  
 p\_creation\_date **IN** **DATE**,  
 p\_created\_by **IN** **NUMBER**,  
 p\_last\_updated\_by **IN** **NUMBER**,  
 p\_last\_update\_date **IN** **DATE**,  
 p\_last\_update\_login **IN** **NUMBER**,  
 p\_program\_application\_id **IN** **NUMBER**,  
 p\_program\_id **IN** **NUMBER**,  
 p\_program\_update\_date **IN** **DATE**,  
 p\_request\_id **IN** **NUMBER**,  
 p\_attribute\_category **IN** **VARCHAR2**,  
 p\_attribute1 **IN** **VARCHAR2**,  
 p\_attribute2 **IN** **VARCHAR2**,  
 p\_attribute3 **IN** **VARCHAR2**,  
 p\_attribute4 **IN** **VARCHAR2**,  
 p\_attribute5 **IN** **VARCHAR2**,  
 p\_attribute6 **IN** **VARCHAR2**,  
 p\_attribute7 **IN** **VARCHAR2**,  
 p\_attribute8 **IN** **VARCHAR2**,  
 p\_attribute9 **IN** **VARCHAR2**,  
 p\_attribute10 **IN** **VARCHAR2**,  
 p\_attribute11 **IN** **VARCHAR2**,  
 p\_attribute12 **IN** **VARCHAR2**,  
 p\_attribute13 **IN** **VARCHAR2**,  
 p\_attribute14 **IN** **VARCHAR2**,  
 p\_attribute15 **IN** **VARCHAR2**)  
   
 **IS**  
 **CURSOR** c **IS**  
 **SELECT** header\_id,  
 org\_id,  
 order\_number,  
 ordered\_date,  
 order\_type,  
 customer\_id,  
 salesrep\_id,  
 price\_list\_id,  
 currency\_code,  
 flow\_status\_code,  
 description,  
 creation\_date,  
 created\_by,  
 last\_updated\_by,  
 last\_update\_date,  
 last\_update\_login,  
 program\_application\_id,  
 program\_id,  
 program\_update\_date,  
 request\_id,  
 attribute\_category,  
 attribute1,  
 attribute2,  
 attribute3,  
 attribute4,  
 attribute5,  
 attribute6,  
 attribute7,  
 attribute8,  
 attribute9,  
 attribute10,  
 attribute11,  
 attribute12,  
 attribute13,  
 attribute14,  
 attribute15  
 **FROM** cux\_order\_headers\_all  
 **WHERE** header\_id = p\_header\_id  
 **FOR** **UPDATE** **OF** header\_id **NOWAIT**;  
   
 rec c%**ROWTYPE**;  
 **BEGIN**  
   
 **OPEN** c;  
 **FETCH** c  
 **INTO** rec;  
 **IF** (c%**NOTFOUND**) **THEN**  
 **CLOSE** c;  
 fnd\_message.set\_name('FND', 'FORM\_RECORD\_DELETED');  
 app\_exception.raise\_exception;  
 **END** **IF**;  
 **CLOSE** c;  
   
 **IF** ((rec.header\_id = p\_header\_id) **AND**  
 ((rec.org\_id = p\_org\_id) **OR**  
 ((rec.org\_id **IS** **NULL**) **AND** (p\_org\_id **IS** **NULL**))) **AND**  
 ((rec.order\_number = p\_order\_number) **OR**  
 ((rec.order\_number **IS** **NULL**) **AND** (p\_order\_number **IS** **NULL**))) **AND**  
 ((rec.ordered\_date = p\_ordered\_date) **OR**  
 ((rec.ordered\_date **IS** **NULL**) **AND** (p\_ordered\_date **IS** **NULL**))) **AND**  
 ((rec.order\_type = p\_order\_type) **OR**  
 ((rec.order\_type **IS** **NULL**) **AND** (p\_order\_type **IS** **NULL**))) **AND**  
 ((rec.customer\_id = p\_customer\_id) **OR**  
 ((rec.customer\_id **IS** **NULL**) **AND** (p\_customer\_id **IS** **NULL**))) **AND**  
 ((rec.salesrep\_id = p\_salesrep\_id) **OR**  
 ((rec.salesrep\_id **IS** **NULL**) **AND** (p\_salesrep\_id **IS** **NULL**))) **AND**  
 ((rec.price\_list\_id = p\_price\_list\_id) **OR**  
 ((rec.price\_list\_id **IS** **NULL**) **AND** (p\_price\_list\_id **IS** **NULL**))) **AND**  
 ((rec.currency\_code = p\_currency\_code) **OR**  
 ((rec.currency\_code **IS** **NULL**) **AND** (p\_currency\_code **IS** **NULL**))) **AND**  
 ((rec.flow\_status\_code = p\_flow\_status\_code) **OR**  
 ((rec.flow\_status\_code **IS** **NULL**) **AND** (p\_flow\_status\_code **IS** **NULL**))) **AND**  
 ((rec.description = p\_description) **OR**  
 ((rec.description **IS** **NULL**) **AND** (p\_description **IS** **NULL**))) **AND**  
 ((rec.creation\_date = p\_creation\_date) **OR**  
 ((rec.creation\_date **IS** **NULL**) **AND** (p\_creation\_date **IS** **NULL**))) **AND**  
 ((rec.created\_by = p\_created\_by) **OR**  
 ((rec.created\_by **IS** **NULL**) **AND** (p\_created\_by **IS** **NULL**))) **AND**  
 ((rec.last\_updated\_by = p\_last\_updated\_by) **OR**  
 ((rec.last\_updated\_by **IS** **NULL**) **AND** (p\_last\_updated\_by **IS** **NULL**))) **AND**  
 ((rec.last\_update\_date = p\_last\_update\_date) **OR**  
 ((rec.last\_update\_date **IS** **NULL**) **AND** (p\_last\_update\_date **IS** **NULL**))) **AND**  
 ((rec.last\_update\_login = p\_last\_update\_login) **OR**  
 ((rec.last\_update\_login **IS** **NULL**) **AND** (p\_last\_update\_login **IS** **NULL**))) **AND**  
 ((rec.program\_application\_id = p\_program\_application\_id) **OR**  
 ((rec.program\_application\_id **IS** **NULL**) **AND**  
 (p\_program\_application\_id **IS** **NULL**))) **AND**  
 ((rec.program\_id = p\_program\_id) **OR**  
 ((rec.program\_id **IS** **NULL**) **AND** (p\_program\_id **IS** **NULL**))) **AND**  
 ((rec.program\_update\_date = p\_program\_update\_date) **OR**  
 ((rec.program\_update\_date **IS** **NULL**) **AND**  
 (p\_program\_update\_date **IS** **NULL**))) **AND**  
 ((rec.request\_id = p\_request\_id) **OR**  
 ((rec.request\_id **IS** **NULL**) **AND** (p\_request\_id **IS** **NULL**))) **AND**  
 ((rec.attribute\_category = p\_attribute\_category) **OR**  
 ((rec.attribute\_category **IS** **NULL**) **AND**  
 (p\_attribute\_category **IS** **NULL**))) **AND**  
 ((rec.attribute1 = p\_attribute1) **OR**  
 ((rec.attribute1 **IS** **NULL**) **AND** (p\_attribute1 **IS** **NULL**))) **AND**  
 ((rec.attribute2 = p\_attribute2) **OR**  
 ((rec.attribute2 **IS** **NULL**) **AND** (p\_attribute2 **IS** **NULL**))) **AND**  
 ((rec.attribute3 = p\_attribute3) **OR**  
 ((rec.attribute3 **IS** **NULL**) **AND** (p\_attribute3 **IS** **NULL**))) **AND**  
 ((rec.attribute4 = p\_attribute4) **OR**  
 ((rec.attribute4 **IS** **NULL**) **AND** (p\_attribute4 **IS** **NULL**))) **AND**  
 ((rec.attribute5 = p\_attribute5) **OR**  
 ((rec.attribute5 **IS** **NULL**) **AND** (p\_attribute5 **IS** **NULL**))) **AND**  
 ((rec.attribute6 = p\_attribute6) **OR**  
 ((rec.attribute6 **IS** **NULL**) **AND** (p\_attribute6 **IS** **NULL**))) **AND**  
 ((rec.attribute7 = p\_attribute7) **OR**  
 ((rec.attribute7 **IS** **NULL**) **AND** (p\_attribute7 **IS** **NULL**))) **AND**  
 ((rec.attribute8 = p\_attribute8) **OR**  
 ((rec.attribute8 **IS** **NULL**) **AND** (p\_attribute8 **IS** **NULL**))) **AND**  
 ((rec.attribute9 = p\_attribute9) **OR**  
 ((rec.attribute9 **IS** **NULL**) **AND** (p\_attribute9 **IS** **NULL**))) **AND**  
 ((rec.attribute10 = p\_attribute10) **OR**  
 ((rec.attribute10 **IS** **NULL**) **AND** (p\_attribute10 **IS** **NULL**))) **AND**  
 ((rec.attribute11 = p\_attribute11) **OR**  
 ((rec.attribute11 **IS** **NULL**) **AND** (p\_attribute11 **IS** **NULL**))) **AND**  
 ((rec.attribute12 = p\_attribute12) **OR**  
 ((rec.attribute12 **IS** **NULL**) **AND** (p\_attribute12 **IS** **NULL**))) **AND**  
 ((rec.attribute13 = p\_attribute13) **OR**  
 ((rec.attribute13 **IS** **NULL**) **AND** (p\_attribute13 **IS** **NULL**))) **AND**  
 ((rec.attribute14 = p\_attribute14) **OR**  
 ((rec.attribute14 **IS** **NULL**) **AND** (p\_attribute14 **IS** **NULL**))) **AND**  
 ((rec.attribute15 = p\_attribute15) **OR**  
 ((rec.attribute15 **IS** **NULL**) **AND** (p\_attribute15 **IS** **NULL**)))) **THEN**  
 **NULL**;  
 **ELSE**  
 fnd\_message.set\_name('FND', 'FORM\_RECORD\_CHANGED');  
 app\_exception.raise\_exception;  
 **END** **IF**;  
   
 **END** lock\_row;  
  
 */\*=====================================  
 \*\* PROCEDURE: update\_row()  
 \*\*=====================================\*/*  
 **PROCEDURE** update\_row(p\_header\_id **IN** **NUMBER**,  
 p\_org\_id **IN** **NUMBER**,  
 p\_order\_number **IN** **NUMBER**,  
 p\_ordered\_date **IN** **DATE**,  
 p\_order\_type **IN** **VARCHAR2**,  
 p\_customer\_id **IN** **NUMBER**,  
 p\_salesrep\_id **IN** **NUMBER** **DEFAULT** **NULL**,  
 p\_price\_list\_id **IN** **NUMBER** **DEFAULT** **NULL**,  
 p\_currency\_code **IN** **VARCHAR2**,  
 p\_flow\_status\_code **IN** **VARCHAR2**,  
 p\_description **IN** **VARCHAR2** **DEFAULT** **NULL**,  
 p\_last\_updated\_by **IN** **NUMBER**,  
 p\_last\_update\_date **IN** **DATE**,  
 p\_last\_update\_login **IN** **NUMBER** **DEFAULT** **NULL**,  
 p\_program\_application\_id **IN** **NUMBER** **DEFAULT** **NULL**,  
 p\_program\_id **IN** **NUMBER** **DEFAULT** **NULL**,  
 p\_program\_update\_date **IN** **DATE** **DEFAULT** **NULL**,  
 p\_request\_id **IN** **NUMBER** **DEFAULT** **NULL**,  
 p\_attribute\_category **IN** **VARCHAR2** **DEFAULT** **NULL**,  
 p\_attribute1 **IN** **VARCHAR2** **DEFAULT** **NULL**,  
 p\_attribute2 **IN** **VARCHAR2** **DEFAULT** **NULL**,  
 p\_attribute3 **IN** **VARCHAR2** **DEFAULT** **NULL**,  
 p\_attribute4 **IN** **VARCHAR2** **DEFAULT** **NULL**,  
 p\_attribute5 **IN** **VARCHAR2** **DEFAULT** **NULL**,  
 p\_attribute6 **IN** **VARCHAR2** **DEFAULT** **NULL**,  
 p\_attribute7 **IN** **VARCHAR2** **DEFAULT** **NULL**,  
 p\_attribute8 **IN** **VARCHAR2** **DEFAULT** **NULL**,  
 p\_attribute9 **IN** **VARCHAR2** **DEFAULT** **NULL**,  
 p\_attribute10 **IN** **VARCHAR2** **DEFAULT** **NULL**,  
 p\_attribute11 **IN** **VARCHAR2** **DEFAULT** **NULL**,  
 p\_attribute12 **IN** **VARCHAR2** **DEFAULT** **NULL**,  
 p\_attribute13 **IN** **VARCHAR2** **DEFAULT** **NULL**,  
 p\_attribute14 **IN** **VARCHAR2** **DEFAULT** **NULL**,  
 p\_attribute15 **IN** **VARCHAR2** **DEFAULT** **NULL**)  
   
 **IS**  
 **BEGIN**  
   
 **UPDATE** cux\_order\_headers\_all  
 **SET** header\_id = p\_header\_id,  
 org\_id = p\_org\_id,  
 order\_number = p\_order\_number,  
 ordered\_date = p\_ordered\_date,  
 order\_type = p\_order\_type,  
 customer\_id = p\_customer\_id,  
 salesrep\_id = p\_salesrep\_id,  
 price\_list\_id = p\_price\_list\_id,  
 currency\_code = p\_currency\_code,  
 flow\_status\_code = p\_flow\_status\_code,  
 description = p\_description,  
 last\_updated\_by = p\_last\_updated\_by,  
 last\_update\_date = p\_last\_update\_date,  
 last\_update\_login = p\_last\_update\_login,  
 program\_application\_id = p\_program\_application\_id,  
 program\_id = p\_program\_id,  
 program\_update\_date = p\_program\_update\_date,  
 request\_id = p\_request\_id,  
 attribute\_category = p\_attribute\_category,  
 attribute1 = p\_attribute1,  
 attribute2 = p\_attribute2,  
 attribute3 = p\_attribute3,  
 attribute4 = p\_attribute4,  
 attribute5 = p\_attribute5,  
 attribute6 = p\_attribute6,  
 attribute7 = p\_attribute7,  
 attribute8 = p\_attribute8,  
 attribute9 = p\_attribute9,  
 attribute10 = p\_attribute10,  
 attribute11 = p\_attribute11,  
 attribute12 = p\_attribute12,  
 attribute13 = p\_attribute13,  
 attribute14 = p\_attribute14,  
 attribute15 = p\_attribute15  
 **WHERE** header\_id = p\_header\_id;  
   
 **IF** (**SQL**%**NOTFOUND**) **THEN**  
 **RAISE** no\_data\_found;  
 **END** **IF**;  
   
 **END** update\_row;  
  
 */\*=====================================  
 \*\* PROCEDURE: delete\_row()  
 \*\*=====================================\*/*  
 **PROCEDURE** delete\_row(p\_header\_id **IN** **NUMBER**)  
   
 **IS**  
 **BEGIN**  
   
 **DELETE** **FROM** cux\_order\_headers\_all **WHERE** header\_id = p\_header\_id;  
   
 **IF** (**SQL**%**NOTFOUND**) **THEN**  
 **RAISE** no\_data\_found;  
 **END** **IF**;  
   
 **END** delete\_row;  
  
**END**;  
/

### 一个简单的例子.从模版开始设计

#### 快速认识Forms Builder开发环境

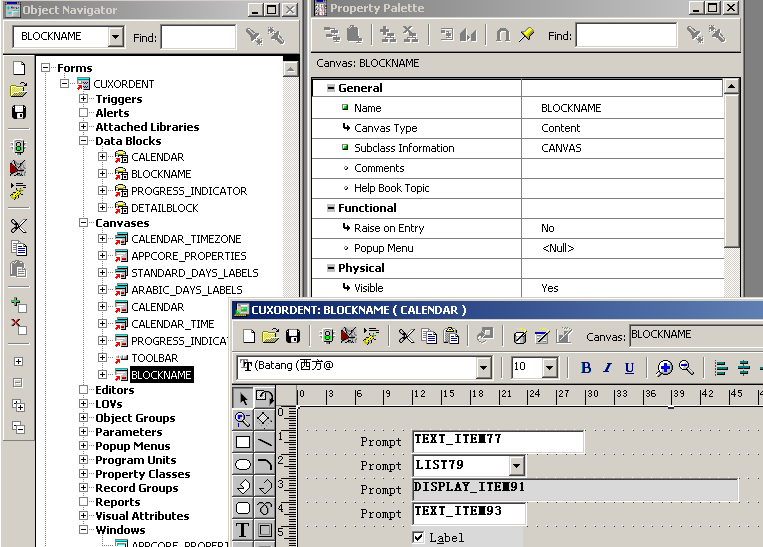
1. Object Navigator，分层次的对象管理，常用的有：Triggers触发器、Data Blocks数据块（其下有Items字段、Triggers触发器）、Canvases画布、LOVs下拉列表、Parameters参数、Program Units程序单元、Record Group记录组、Windows窗体。

这些对象可以通过左边红框标记的按钮进行添加、删除。

1. Property Palette，分类别的属性设置，主要属性后面用到再介绍。
2. Canvases，设计各字段在画布上的布局。

快捷键F4：选中Object Navigator中的某个对象，F4可以调出Property Palette。

1. Code Editor代码编辑器：编写PL/SQL代码的地方。



#### 拷贝TEMPLATE.fmb

拷贝TEMPLATE.fmb，改名为CUXORDENT.fmb，打开之后把Form Name也改为CUXORDENT，一定要保持一致性。

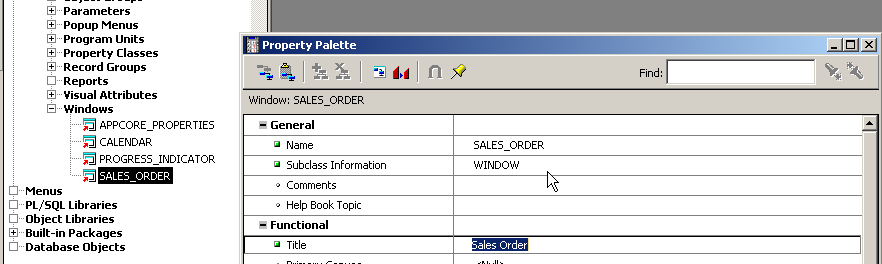
#### 删除多余对象

删除Data Blocks下的两个块——BLOCKNAME、DETAILBLOCK，它们是模版自带的示例主从块。

删除Canvases下的一个画布——BLOCKNAME。

#### 修改Windows名称

修改Windows下的BLOCKNAME这个Window，在Property Palette中将Name改为“**SALES\_ORDER**”，将Title改为“**Sales Order**”。



#### 修改2个触发器、1个程序单元

因为一个Forms有多个Window，所以需要明确指出哪个Window是主Window，这样在关闭主Window的时候将退出整个Forms；此外，我们也须要在源代码中记录作者、开发时间、版本等备注信息。这些是通过代码而非属性来控制的。

1. 修改Forms级触发器PRE-FORM

将其中的app\_window.set\_window\_position('BLOCKNAME', 'FIRST\_WINDOW');改为app\_window.set\_window\_position('SALES\_ORDER', 'FIRST\_WINDOW');

同时修改上面几行代码中的作者、时间、版本、Form描述、应用简称，示例如下：

FND\_STANDARD.FORM\_INFO('$Revision: 1.0.0 $', 'Sales Order Form', 'SCF',  
 '$Date: 2007/02/28 11:02 $', '$Author: huajhua $');

1. 修改Forms级触发器WHEN-NEW-FORM-INSTANCE

主要修改Form名字、版本和日期。

1. 修改Program Unit下app\_custom中的close\_window过程

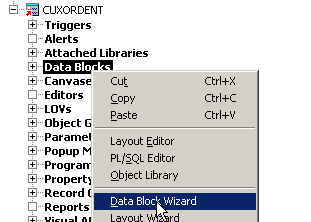
将**if** (wnd = '<your first window>') **then** app\_window.close\_first\_window;改为

**if** (wnd = 'SALES\_ORDER') **then** app\_window.close\_first\_window;

#### 创建Block数据块

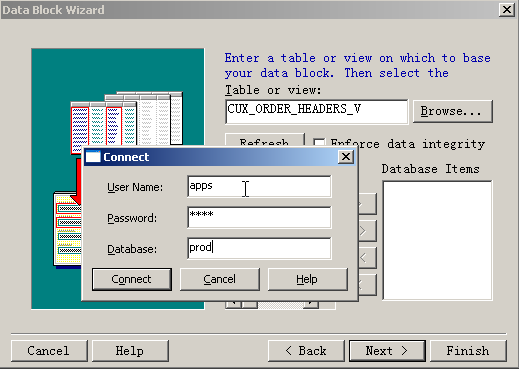
数据块，定义Form上的字段与数据库中的字段是如何对应的，同时定义块和字段的各种操作特性——字段类型、长度、默认值、可否增删改查等等。

N: 在Data Blocks上右键，选择数据块创建向导：



之后第一步跳过，第二步保持默认的选择“Table or View”

第三步输入View名字CUX\_ORDER\_HEADERS\_V，如果没有登录过，则会弹出登录框：



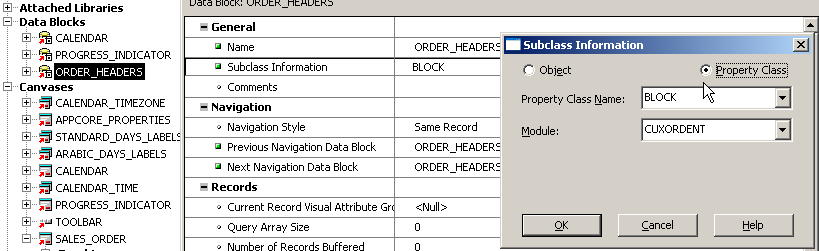
登录后Available Column将显示所有视图字段，点击>>全部选过来。

第四步跳过，第五步选择“Just create the data block”后Finish。

Forms Builder将自动将表字段的名称、类型、长度、是否必须等信息带到Block下的Items中。

#### 设置Block属性及其Subclass

默认Block名字为视图名字，需要“精简”，此例改为“ORDER\_HEADERS”，同时需要设置块属性中的Subclass为Block：



此外，需要设置块的前后导航属性，本例目前仅有一个有意义的数据块，这里将Previous和Next Navigation Data Block均设置为自身，这样在运行时，使用快捷键Shift+Pageup和Shift+Pagedown时，均不会跳离本块。

子类：类似Java中的子类，用来继承Item类型和各种属性，在开发过程中，要严格设置各种Item的子类，不得手工随意修改对象的属性，从而保持界面有一致的风格；设置过子类的对象，图标上有个红色的箭头。

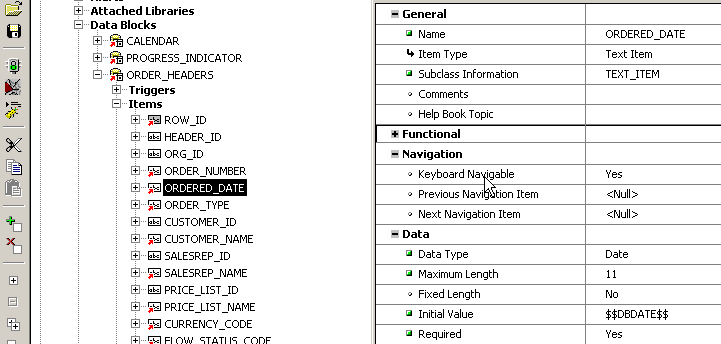
#### 设置Item属性及其Subclass

设置需要显示Item和特殊Item的子类，同时设置部分Item的默认值：

| Item | Subclass | Required | Initial Value | 说明 |
| --- | --- | --- | --- | --- |
|  |  |  |  |  |
| ROW\_ID | ROW\_ID |  |  | 不显示 |
| ORDER\_NUMBER | TEXT\_ITEM | Yes |  |  |
| ORDERED\_DATE | TEXT\_ITEM | Yes | $$DBDATE$$ | 系统日期 |
| ORDER\_TYPE | TEXT\_ITEM | Yes | I |  |
| CUSTOMER\_NAME | TEXT\_ITEM | Yes |  |  |
| SALESREP\_NAME | TEXT\_ITEM | No |  |  |
| PRICE\_LIST\_NAME | TEXT\_ITEM | No |  |  |
| CURRENCY\_CODE | TEXT\_ITEM | Yes |  |  |
| FLOW\_STATUS\_CODE | TEXT\_ITEM | Yes | ENTERED |  |
| DESCRIPTION | TEXT\_ITEM | No |  |  |

此外，为测试和学习方便，这里先把ORG\_ID和CUSTOMER\_ID的默认值设置为0。

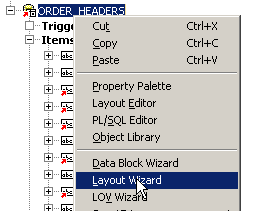
以ORDERED\_DATE为例，设置画面如下：



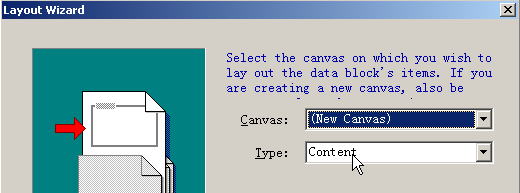
#### 创建Canvas画布

画布，用来设计各字段的布局，画布必需放置在Window上后才真正对用户“可见”。最基本的画布为Content画布，一个Window必需有且仅有一个Content画布。

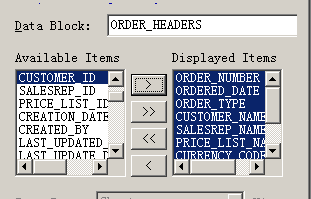
N: 在Data Blocks上右键，选择布局创建向导：



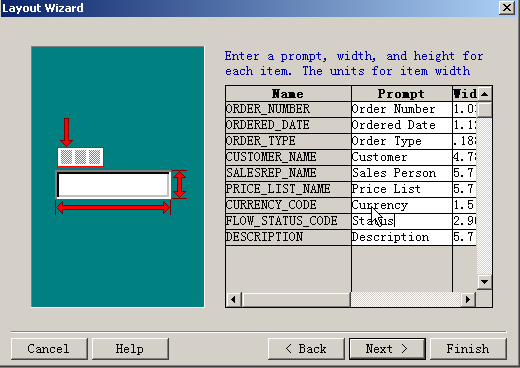
第一步选择New Canvas，Type选择Content：



第二步选择要显示的字段：



第三步修改各字段的提示和显示长度：

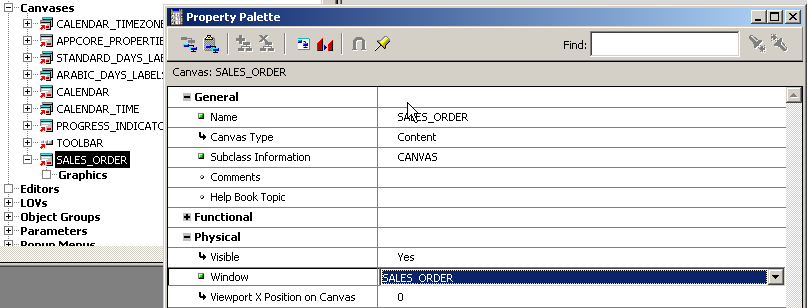


后面两步跳过直至Finish。

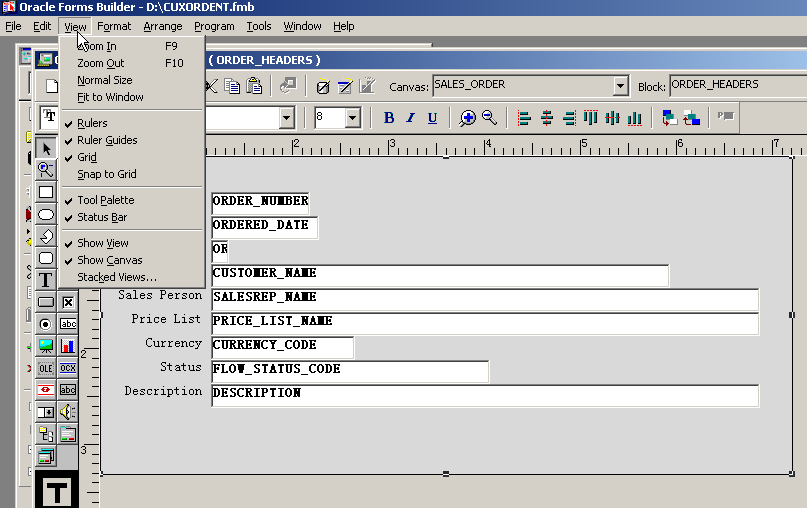
在打开的画布设计界面，删除自动生成的Frame。

#### 设置画布属性和子类、调整布局

1、设置画布名字为SALES\_ORDER，放置的Window为SALES\_ORDER，子类为Canvas：



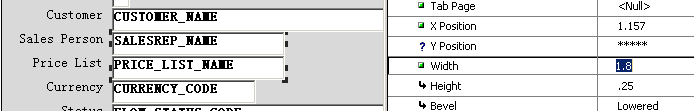
2、调整View和Canvas大小，Content画布这两者设置为一样大，即拉到重叠为止。



Canvas与View：Canvas是整个画布的大小，Item是放置在Canvas上的；View是这个画布的可见部分，如果View小于Canvas，那么通常需要借助滚动条来查看整个Canvas中的内容；可以通过菜单View中的Show View、Show Canvas来确定哪个框是Canvas哪个框是View。

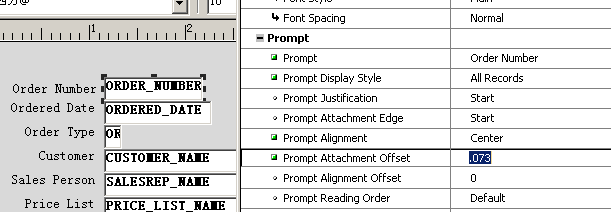
#### 调整布局

调整默认的布局，这里通过属性直接将Sales Person和Price List的宽度设置为1.8：



#### 调整Prompt提示

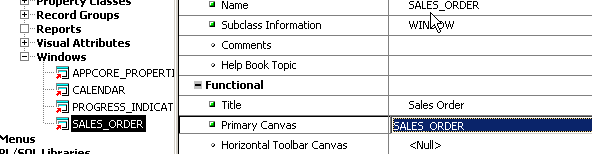
Oracle标准的Form中，Form形式的布局，提示放置在字段左边，居中对齐，并且距离字段0.073；而Table形式的布局，提示放置在第一行字段上边，数字靠右，其他靠左，并分别距离字段边缘0.05。以ORDER\_NUMBER为例设置如下：



其他字段可以按住Ctrl一起选中，然后一次性设置。

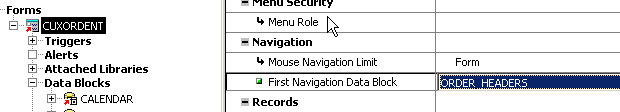
#### 设置Windows属性

设置Window“SALES\_ORDER”的Primary Canvas为“SALES\_ORDER”，该Window的大小将自动调整为Content Canvas的大小：



#### 设置Form属性

设置Form的第一个导航块为ORDER\_HEADERS，这样运行时，一进入Form，光标将停在该块的第一个字段上：

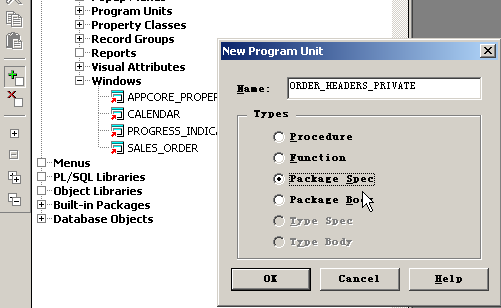


### 一个简单的例子.编写数据操作触发器

#### 编写数据操作Program Unit

该Program Unit主要用来调用之前步骤创建的数据库API。

1. 选中对象管理器中的Program Unit，点击左边的+，选择Package Spec，在Name中输入ORDER\_HEADERS\_PRIVATE：



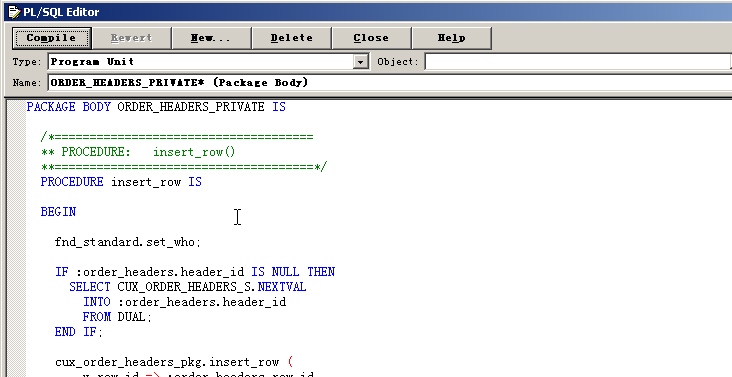
1. OK后，输入如下包头代码：

**PACKAGE** ORDER\_HEADERS\_PRIVATE **IS**  
  
 **PROCEDURE** insert\_row;  
 **PROCEDURE** lock\_row;  
 **PROCEDURE** update\_row;  
 **PROCEDURE** delete\_row;  
  
**END** ORDER\_HEADERS\_PRIVATE;

1. 同1，选择Package Body，OK后输入如下包体代码，注意块字段的值的引用格式“:块名.字段名”：

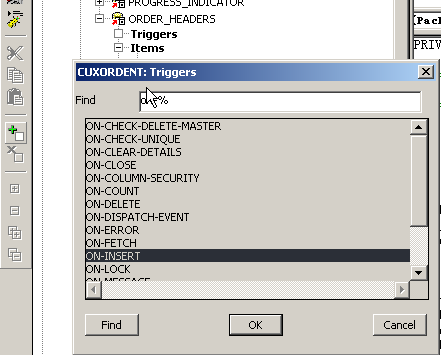
**PACKAGE** **BODY** ORDER\_HEADERS\_PRIVATE **IS**  
  
 */\*=====================================  
 \*\* PROCEDURE: insert\_row()  
 \*\*=====================================\*/*  
 **PROCEDURE** insert\_row **IS**  
  
 **BEGIN**  
  
 fnd\_standard.set\_who;  
  
 **IF** :order\_headers.header\_id **IS** **NULL** **THEN**  
 **SELECT** CUX\_ORDER\_HEADERS\_S.**NEXTVAL**  
 **INTO** :order\_headers.header\_id  
 **FROM** DUAL;  
 **END** **IF**;  
  
 cux\_order\_headers\_pkg.insert\_row (  
 x\_row\_id => :order\_headers.row\_id  
 ,x\_header\_id => :order\_headers.header\_id  
 ,p\_org\_id => :order\_headers.org\_id  
 ,p\_order\_number => :order\_headers.order\_number  
 ,p\_ordered\_date => :order\_headers.ordered\_date  
 ,p\_order\_type => :order\_headers.order\_type  
 ,p\_customer\_id => :order\_headers.customer\_id  
 ,p\_salesrep\_id => :order\_headers.salesrep\_id  
 ,p\_price\_list\_id => :order\_headers.price\_list\_id  
 ,p\_currency\_code => :order\_headers.currency\_code  
 ,p\_flow\_status\_code => :order\_headers.flow\_status\_code  
 ,p\_description => :order\_headers.**description**  
 ,p\_creation\_date => :order\_headers.creation\_date  
 ,p\_created\_by => :order\_headers.created\_by  
 ,p\_last\_updated\_by => :order\_headers.last\_updated\_by  
 ,p\_last\_update\_date => :order\_headers.last\_update\_date  
 ,p\_last\_update\_login => :order\_headers.last\_update\_login  
 ,p\_program\_application\_id => :order\_headers.program\_application\_id  
 ,p\_program\_id => :order\_headers.program\_id  
 ,p\_program\_update\_date => :order\_headers.program\_update\_date  
 ,p\_request\_id => :order\_headers.request\_id  
 ,p\_attribute\_category => :order\_headers.attribute\_category  
 ,p\_attribute1 => :order\_headers.attribute1  
 ,p\_attribute2 => :order\_headers.attribute2  
 ,p\_attribute3 => :order\_headers.attribute3  
 ,p\_attribute4 => :order\_headers.attribute4  
 ,p\_attribute5 => :order\_headers.attribute5  
 ,p\_attribute6 => :order\_headers.attribute6  
 ,p\_attribute7 => :order\_headers.attribute7  
 ,p\_attribute8 => :order\_headers.attribute8  
 ,p\_attribute9 => :order\_headers.attribute9  
 ,p\_attribute10 => :order\_headers.attribute10  
 ,p\_attribute11 => :order\_headers.attribute11  
 ,p\_attribute12 => :order\_headers.attribute12  
 ,p\_attribute13 => :order\_headers.attribute13  
 ,p\_attribute14 => :order\_headers.attribute14  
 ,p\_attribute15 => :order\_headers.attribute15  
 );  
 **END** insert\_row;  
  
 */\*=====================================  
 \*\* PROCEDURE: lock\_row()  
 \*\*=====================================\*/*  
 **PROCEDURE** lock\_row **IS**  
  
 i **NUMBER** := 0;  
 **BEGIN**  
  
 **LOOP**  
 **BEGIN**  
 i := i + 1;  
 cux\_order\_headers\_pkg.lock\_row(  
 p\_header\_id => :order\_headers.header\_id  
 ,p\_org\_id => :order\_headers.org\_id  
 ,p\_order\_number => :order\_headers.order\_number  
 ,p\_ordered\_date => :order\_headers.ordered\_date  
 ,p\_order\_type => :order\_headers.order\_type  
 ,p\_customer\_id => :order\_headers.customer\_id  
 ,p\_salesrep\_id => :order\_headers.salesrep\_id  
 ,p\_price\_list\_id => :order\_headers.price\_list\_id  
 ,p\_currency\_code => :order\_headers.currency\_code  
 ,p\_flow\_status\_code => :order\_headers.flow\_status\_code  
 ,p\_description => :order\_headers.**description**  
 ,p\_creation\_date => :order\_headers.creation\_date  
 ,p\_created\_by => :order\_headers.created\_by  
 ,p\_last\_updated\_by => :order\_headers.last\_updated\_by  
 ,p\_last\_update\_date => :order\_headers.last\_update\_date  
 ,p\_last\_update\_login => :order\_headers.last\_update\_login  
 ,p\_program\_application\_id => :order\_headers.program\_application\_id  
 ,p\_program\_id => :order\_headers.program\_id  
 ,p\_program\_update\_date => :order\_headers.program\_update\_date  
 ,p\_request\_id => :order\_headers.request\_id  
 ,p\_attribute\_category => :order\_headers.attribute\_category  
 ,p\_attribute1 => :order\_headers.attribute1  
 ,p\_attribute2 => :order\_headers.attribute2  
 ,p\_attribute3 => :order\_headers.attribute3  
 ,p\_attribute4 => :order\_headers.attribute4  
 ,p\_attribute5 => :order\_headers.attribute5  
 ,p\_attribute6 => :order\_headers.attribute6  
 ,p\_attribute7 => :order\_headers.attribute7  
 ,p\_attribute8 => :order\_headers.attribute8  
 ,p\_attribute9 => :order\_headers.attribute9  
 ,p\_attribute10 => :order\_headers.attribute10  
 ,p\_attribute11 => :order\_headers.attribute11  
 ,p\_attribute12 => :order\_headers.attribute12  
 ,p\_attribute13 => :order\_headers.attribute13  
 ,p\_attribute14 => :order\_headers.attribute14  
 ,p\_attribute15 => :order\_headers.attribute15  
 );  
 **RETURN**;  
 **EXCEPTION**  
 **WHEN** app\_exception.record\_lock\_exception **THEN**  
 app\_exception.record\_lock\_error(i);  
 **END**;  
 **END** **LOOP**;  
  
 **END** lock\_row;  
  
 */\*=====================================  
 \*\* PROCEDURE: update\_row()  
 \*\*=====================================\*/*  
 **PROCEDURE** update\_row **IS**  
 **BEGIN**  
  
 fnd\_standard.set\_who;  
  
 cux\_order\_headers\_pkg.update\_row(  
 p\_header\_id => :order\_headers.header\_id  
 ,p\_org\_id => :order\_headers.org\_id  
 ,p\_order\_number => :order\_headers.order\_number  
 ,p\_ordered\_date => :order\_headers.ordered\_date  
 ,p\_order\_type => :order\_headers.order\_type  
 ,p\_customer\_id => :order\_headers.customer\_id  
 ,p\_salesrep\_id => :order\_headers.salesrep\_id  
 ,p\_price\_list\_id => :order\_headers.price\_list\_id  
 ,p\_currency\_code => :order\_headers.currency\_code  
 ,p\_flow\_status\_code => :order\_headers.flow\_status\_code  
 ,p\_description => :order\_headers.**description**  
 ,p\_last\_updated\_by => :order\_headers.last\_updated\_by  
 ,p\_last\_update\_date => :order\_headers.last\_update\_date  
 ,p\_last\_update\_login => :order\_headers.last\_update\_login  
 ,p\_program\_application\_id => :order\_headers.program\_application\_id  
 ,p\_program\_id => :order\_headers.program\_id  
 ,p\_program\_update\_date => :order\_headers.program\_update\_date  
 ,p\_request\_id => :order\_headers.request\_id  
 ,p\_attribute\_category => :order\_headers.attribute\_category  
 ,p\_attribute1 => :order\_headers.attribute1  
 ,p\_attribute2 => :order\_headers.attribute2  
 ,p\_attribute3 => :order\_headers.attribute3  
 ,p\_attribute4 => :order\_headers.attribute4  
 ,p\_attribute5 => :order\_headers.attribute5  
 ,p\_attribute6 => :order\_headers.attribute6  
 ,p\_attribute7 => :order\_headers.attribute7  
 ,p\_attribute8 => :order\_headers.attribute8  
 ,p\_attribute9 => :order\_headers.attribute9  
 ,p\_attribute10 => :order\_headers.attribute10  
 ,p\_attribute11 => :order\_headers.attribute11  
 ,p\_attribute12 => :order\_headers.attribute12  
 ,p\_attribute13 => :order\_headers.attribute13  
 ,p\_attribute14 => :order\_headers.attribute14  
 ,p\_attribute15 => :order\_headers.attribute15  
 );  
 **END** update\_row;  
  
 */\*=====================================  
 \*\* PROCEDURE: delete\_row()  
 \*\*=====================================\*/*  
 **PROCEDURE** delete\_row **IS**  
 **BEGIN**  
  
 cux\_order\_headers\_pkg.delete\_row(  
 p\_header\_id => :order\_headers.header\_id  
 );  
 **END** delete\_row;  
  
**END** ORDER\_HEADERS\_PRIVATE;

1. 点击PL/SQL Editor窗口顶上的**Compile按钮**，必须保证没有错误：



#### 编写块的On触发器

1. 选中Block“ORDER\_HEADERS”下的Triggers，点击左边的+，选择ON-INSERT触发器：



1. OK后输入如下代码：

ORDER\_HEADERS\_PRIVATE.insert\_row;

1. 同1，选择ON-LOCK后输入如下代码：

ORDER\_HEADERS\_PRIVATE.lock\_row;

1. 同1，选择ON-UPDATE后输入如下代码：

ORDER\_HEADERS\_PRIVATE.update\_row;

1. 同1，选择ON-DELETE后输入如下代码：

ORDER\_HEADERS\_PRIVATE.delete\_row;

理解：对于基于非单表视图的Block来说，这里的4个触发器是必须要写的，具体代码可以直接写在触发器内，但为了模块化管理和今后维护方便，这里分为三层调用，触发器中调用Program Unit中的过程，Program Unit中调用数据库Package中的过程。

### 一个简单的例子.上传&编译

#### 步骤

1. 以**应用OS用户**用**telnet工具**登录服务器，使用如下命令获得要上传的路径：

echo $SCF\_TOP/forms/US

假如得到/d1/oracle/prodappl/scf/1.0.0/forms/US

1. 以**应用OS用户**用**FTP工具**登录服务器，使用如下命令获得要上传的路径，将CUXORDENT.fmb以**二进制方式**上传到/d1/oracle/prodappl/scf/1.0.0/forms。
2. 回到**telnet工具**，使用如下命令，必须进入Forms源文件目录：

cd $AU\_TOP/forms/US

1. 编译Form的命令如下，第一个用于EBS11i（6i版本），第二个用于（10g版本）：

f60gen $SCF\_TOP/forms/US/CUXORDENT apps/password output\_file=$SCF\_TOP/forms/US/CUXORDENT

frmcmp\_batch $SCF\_TOP/forms/US/CUXORDENT apps/password output\_file=$SCF\_TOP/forms/US/CUXORDENT

成功编译的话，最后一行将显示：

Created form file /d1/oracle/prodappl/scf/1.0.0/forms/US/CUXORDENT.fmx

另注：Window版本的EBS，直接在客户端Forms Builder中按CTRL+R，当然肯定无法直接运行，需要我们不断按Esc键取消运行时错误信息（非编译错误！），不过可以生成fmx文件，将其直接拷贝到服务器的 %SCF\_TOP%/forms/US目录下。

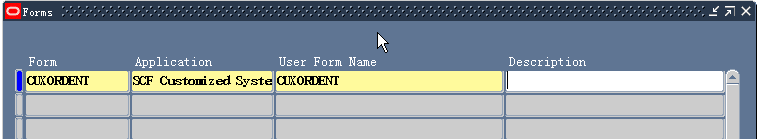
### 一个简单的例子.在EBS中注册运行

#### 登录EBS

用至少需要有Application Developer职责的用户登录，比如Sysadmin用户，然后选择Application Developer职责。

#### 注册Form

N: DEV/Application/Form

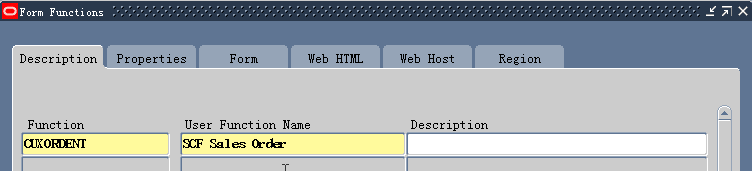


1. 这里的Form、源文件中Form名字、文件名三者要一致，这里是CUXORDENT。
2. 选择合适的Application，通常用企业的客户化应用，这里是SCF Customized System。
3. User Form Name可以输入一个友好的名字，建议直接用Form名字。

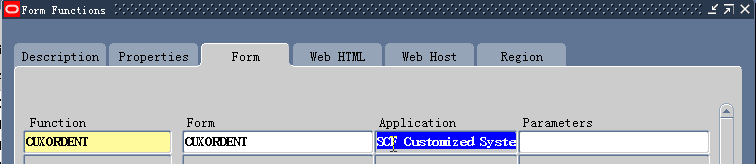
#### 定义Function

N: DEV/Application/Function

1、输入Function名字和Form名字一致，输入一个友好的名字SCF Sales Order



2、切换到Form标签页，选择刚才定义的Form后按Ctrl+S保存。

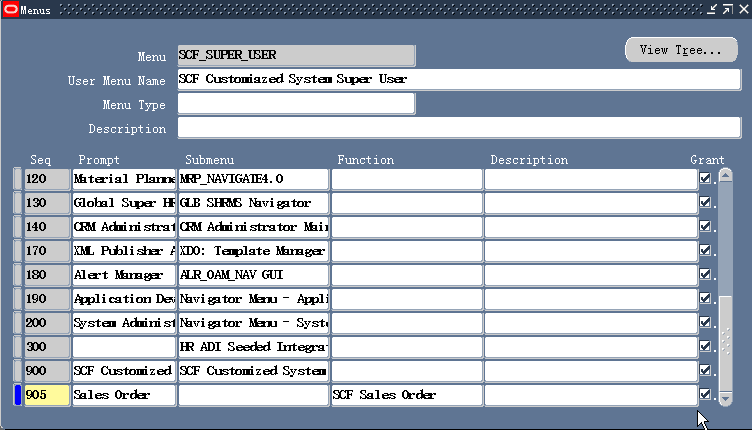


#### 加入Menu

N: DEV/Application/Menu

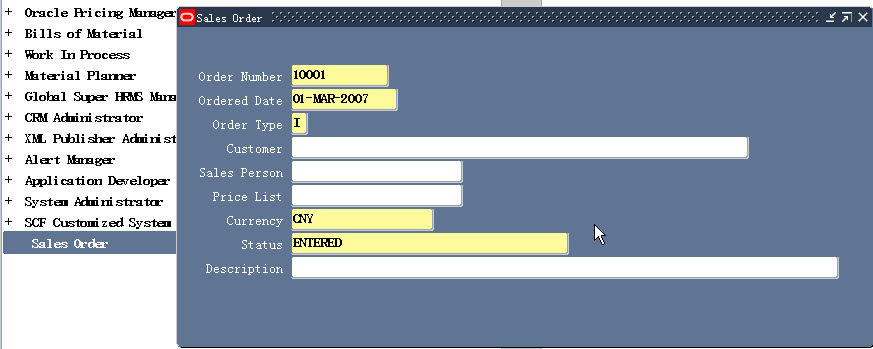
假定我们用SCF Customized System Super User这个职责来测试，该职责对应的菜单为SCF\_SUPER\_USER。

查出菜单SCF\_SUPER\_USER，在下面添加一行：



#### 运行Form

切换到SCF Customized System Super User职责，选择Sales Order菜单，可以看到做好的Form：



因为2个隐藏的必需字段我们暂时没有赋值，而是给了无意义的默认值0，其他可见字段也未做任何有效性控制，所以目前该Form还不能进行实际数据录入操作，不过还是可以测试下打开、字段导航、录入记录、保存记录、关闭等正常操作；但查询记录、修改记录则没法做，暂时只能用SQL直接从表cux\_order\_headers\_all中查看。

字段导航：运行时用Tab或Shift+Tab键可以在字段间导航，默认情况下，导航的顺序与我们开发时块中Item的排列顺序一致（并非和布局中的顺序一致）；我们可以设置Item的属性Keyboard Navigable为No来禁止键盘导航，也可以设置Previous/Next Navigation Item来强制改变前后导航顺序。

截止目前为止的Form源代码请参见CUXORDENT.01.fmb，这个例子最重要，必需保证能够自己做出来；本文也没有面面俱到的详细介绍，还需读者耐心领会。