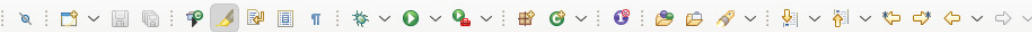


File Edit Source Refactor Navigate Search Project Run Window Help



Package Explorer Project Explorer

```

> jdomainapp [jdomainapp main]
  > jda-common (in common) [jdomainapp main]
  > jda-dcsl (in dcsl) [jdomainapp main]
  > jda-examples-root (in examples) [jdomainapp main]
  > jda-main (in main) [jdomainapp main]
  > jda-module-root (in modules) [jdomainapp main]
    > jda-dcsltool (in dcsltool) [jdomainapp main]
    > jda-dompattern (in dompattern) [jdomainapp main]
    > jda-mbsl (in mbsl) [jdomainapp main]
    > jda-mccl (in mccl) [jdomainapp main]
    > jda-mccltool (in mccltool) [jdomainapp main]
    > jda-mdsa (in mdsa) [jdomainapp main]
    > jda-mosar (in mosar) [jdomainapp main]
    > jda-mscommon (in mscommon) [jdomainapp main]
    > jda-sctl (in sctl) [jdomainapp main]
    > jda-sctltool (in sctltool) [jdomainapp main]
    > jda-swttool (in swttool) [jdomainapp main]
    > pom.xml
    > ReadMe.md
  > local-maven-repo
    > pom.xml
    > ReadMe.md
  > orderman [orderman main]
    > src/main/java
      > org.jda.example.orderman.modules.customer
      > org.jda.example.orderman.modules.customer.model
      > org.jda.example.orderman.modules.delivery
      > org.jda.example.orderman.modules.delivery.model
      > org.jda.example.orderman.modules.fillorder
      > org.jda.example.orderman.modules.fillorder.model
      > org.jda.example.orderman.modules.handleorder
        > ModuleHandleOrder.java
      > org.jda.example.orderman.modules.handleorder.control
      > org.jda.example.orderman.modules.handleorder.control
      > org.jda.example.orderman.modules.handleorder.model
        > HandleOrder.java

```

HandleOrder.java

```

1 package org.jda.example.orderman.modules.handleorder
2
3 import static jda.modules.mbsl.model.graph.NodeT
4
5
6
7
8
9
10
11
12
13
14
15
16
17
18
19
20
21
22
23
24
25
26
27
28
29
30
31
32
33
34
35
36
37
38
39
40
41
42
43
44
45
46
47
48
49
50
51 * @overview
52 /**Activity graph configuration */
53 @AGraph(nodes={
54     /* 1 */
55     @ANode(label="1:CustOrder", zone="top",init=true,
56         refCls=CustOrder.class, serviceCls=DataCon
57         outNodes={ "2:AcceptOrNot"},
58         actSeq={
59             // create new and wait until a new objec
60             @MACT(actName=newObject, endStates={Crea
61             },
62             /* 2: decisional */
63             @ANode(label="2:AcceptOrNot",
64                 refCls=AcceptOrNot.class, nodeType=Decisio
65                 outNodes= {"3:FillOrder", "4:EndOrder"}),
66             /* 3 */
67             @ANode(label="3:FillOrder", zone="top", refCls=F
68                 outNodes= {"5:CustOrder"},
69                 nodeType=Coordinator,
70                 actSeq = {
71                     @MACT(actName=newObject, endStates={NewObj
72                     @MACT(actName=setDataFieldValues, attribNa
73                     @MACT(actName=createObject, endStates={Cre
74                     },
75                     /* 4 */
76                     @ANode(label="4:EndOrder", refCls=EndOrder.class
77                     nodeType=Merge, outNodes= {"7:CustOrder"}))
78                 /* 5 */
79                 @ANode(label="5:CustOrder", zone="3:FillOrder",r
80                 outNodes = {"6:Delivery"},
81                 actSeq = {
82                     @MACT(actName=showObject, endStates = {Upd
83                     },
84                     /* 6: Fork */
85                     @ANode(label="6:Delivery", zone="3:FillOrder",re
86                     nodeType=Fork, outNodes= {"8:CollectPaymen
87                     actSeq = {
88                     @MACT(actName=newObject, endStates={NewObj
89                     @MACT(actName=createObject, endStates={Cre
90                     },
91                     /* 7 */
92                     @ANode(label="7:CustOrder",zone="6:Delivery", re
93                     actSeq={
94                     @MACT(actName=activateView),
95                     @MACT(actName=showObject),
96
97
98
99
100

```

ModuleHandleOrd x SCCOrderMan.jav

```

1 package org.jda.example.orderman.modules.handleorder
2
3 import static jda.modules.mccl.conceptmodel.view.Reg
4
5
6
7
8
9
10
11
12
13
14
15
16
17
18
19
20
21
22
23
24
25
26
27
28
29
30
31
32
33
34
35
36
37
38
39
40
41
42
43
44
45
46
47
48
49
50
51 @ModuleDescriptor(
52     name="ModuleHandleOrder",
53     modelDesc=@ModelDesc( model = HandleOrder.class
54     viewDesc=@ViewDesc(
55         formTitle="Handle customer orders",
56         domainClassLabel="Order Handling", imageIcon
57         view=View.class, viewType=RegionType.Data,
58         layoutBuilderType=DecisionalLayoutBuilder.cl
59         topX=0.5,topY=0.0,widthRatio = 0.5f, heightR
60         excludeComponents={/*general actions*/Export
61             Open, Update, Delete, /*New,*/ First, Prev
62     }
63 ),
64     controllerDesc=@ControllerDesc(
65         controller=Controller.class
66         /*customise createNew command to execute the
67         ,props = {
68         @PropertyDesc(name=PropertyName.controller,
69     }
70 )
71 ,containmentTree=@CTree(
72     root=HandleOrder.class
73     ,edges = {
74         @CEdge(parent=FillOrder.class, child=Cus
75         scopeDesc = @ScopeDesc(stateScope =
76         , @CEdge(parent=CollectPayment.class, ch
77         scopeDesc = @ScopeDesc(stateScope =
78         ,@CEdge(parent=AcceptPayment.class, chil
79         scopeDesc = @ScopeDesc(stateScope =
80         ,@CEdge(parent=Delivery.class, child=Cus
81         scopeDesc = @ScopeDesc(stateScope =
82     }
83 )
84 ,isPrimary=true, childModules={ModuleCustOrder.c
85 )
86 public class ModuleHandleOrder {
87     @AttributeDesc(label="Order Handling")
88     private String title;
89     @AttributeDesc(label="Receive order", type=Default
90     layoutBuilderType=TwoColumnLayoutBuilder.class
91     ,styleField=StyleName.DefaultOnLightYellow
92     ,controllerDesc=@ControllerDesc( openPolicy=Op
93     private Collection<CustOrder> orders;
94     @AttributeDesc(label="Fill Order", type=DefaultPan
95     private Collection<FillOrder> fillOrders;
96 }

```