ABAP OBJECTS – BASICS GUIDED TUTORIAL

Exercises / Solutions by Peter McNulty, Horst Keller / SAP



Introduction

This exercise is a tutorial or guided tour through the fundamental language elements of ABAP Objects and the usage of the respective ABAP Workbench tools. The tutorial is designed for developers who have had little or no experience with ABAP and ABAP Objects until now.



The following class diagram shows the scope of the exercise.

(After finishing the exercise, you should be able to display this diagram for your own classes).



Exercise 1, Classes and Objects (Solution in Package Z_ABAP_OBJECTS_INTRODUCTION_A)

Create a vehicle class

Create a class ZCL_VEHICLE_XX (where XX is your group number).

- The class should have the protected instance attributes SPEED and MAX_SPEED for its speed and maximum speed, and the public methods SPEED_UP, STOP, and SHOW. Furthermore there should be a private attribute that contains a running number for an object ID.
- SPEED_UP should have an IMPORTING parameter STEP. The method should increase the speed by STEP, but not allow it to exceed the maximum speed.
- STOP should reset the speed to zero.
- WRITE should display a message containing the speed and the maximum speed.

Solution

- 1. Logon to the system and open the Object Navigator of the ABAP Workbench (Transaction SE80, enter /nSE80 in the command field of the system task bar).
- 2. Select Local Objects in order to work in a test package that is not transported to other systems.



Hit Enter.



3. Right Click the name of the local package and navigate to the creation of a class (where XX is your group number).

Local Objects	<u> </u>					
ABAP_00_XX	▼ 63°					
Object Name	Descripti					
\$TMP ABAP_OO_XX	Create		Development Coordination			
	Ohenne	, i		,		
	Cnange		Program			
	<u>D</u> isplay	•	Function <u>G</u> roup			
	С <u>о</u> ру		Web Dynpro	•		
	Delete		Dictionary Object	•		
	<u>C</u> heck	Þ	<u>C</u> lass Library	•	Class	
	Display Object Directory Er	ntry	Enterprise Service		Interface	~
	Write Transport Entry		<u>B</u> SP Library	•		
	 Wihara-Head List		<u>E</u> nhancement	×		
			Business Engineering	×		
		,	– Form Object	×		
			Test Object	×		
			Other (1)			
			Other (7)			

4. Fill the pop-up as follows ZCL_VEHICLE_XX (where XX is your group number) and select **Save**.

🔄 Create Class		\times				
Class	ZCL_VEHICLE_XX	수				
Description	Vehicle class					
Instantiation	Public 📋					
Class Type						
🖲 Usual ABAP C	lass					
O Exception Clas	ss					
🗖 With Message Class						
O Persistent class						
O Test Class (ABAP Unit)						
🗌 Final						
Save X						
(Don't forget to un	icheck Final)					



5. Acknowledge the following window without changes (select either Save or Local Object).

🔄 Create Object Directory Entry	× _
Object R3TR CLAS ZCL_V	EHICLE_XX
Attributes	
Package	<u>\$TMP</u>
Person Responsible	ABAP OO XX
Original System Original language Created On	M30 EN English
Local Object 🙎 Lock Ov	verview 🔁 🗶

The same holds for all other development objects during this exercise.

6. Now you enter the Class Builder

C Class Edit Goto Utilities Environment	System <u>H</u> elp								- • × S	AP
	🙆 🚷 I 🖨 🛗 🛙	8 8 10 -	🗘 🕄 🗦	8 🖉	2 🖪					
Class Builder: Change Class ZCL_VEHICLE_XX										
	2 II 7 I	📘 🔙 Local	Definitions	/Impler	mentation	s 🗋 Constr	uctor 📘	Class cons	structor 🛛 🕄 S	ource Code-Based
MIME Repository	Class Interface	ZCL_	VEHICLE_>	Х		Impleme	ented / Inacti	ive		
뤕Repository Browser	Properties	Interfaces	Friends	s A	.ttributes	Methods	Events	Types	Aliases	
निः Repository Information System										
E Tag Browser	Parameter	🖏 Exceptior		€		X 🖻 🛱 🛛	当日日	2	🗌 Filter	
异 Transport Organizer	Method		Level Vi	si M	Descrip	tion				
Example 1 Test Repository										
Local Objects										
				_						
				_	_					
Class Library				_						
					- I					
▷ CL_VEHICLE_XX Veh										
							Þ	M30 (1) 8	800 🖪 iwdfvm30)22 INS 🎽 🎢

Here you can edit the class either in Form-Based mode (default) or in Source Code-Based mode. Use the respective button to toggle between the modes.



7. Switch to Source Code-Based mode existing template with the following code (where XX is your group number).

CLASS zcl_vehicle_xx DEFINITION PUBLIC CREATE PUBLIC. PUBLIC SECTION. METHODS constructor. METHODS speed_up IMPORTING



```
step TYPE i.
     METHODS stop.
     METHODS show.
   PROTECTED SECTION.
     DATA: speed TYPE i,
           max speed TYPE i VALUE 50.
   PRIVATE SECTION.
     DATA id TYPE i .
     CLASS-DATA object count TYPE i.
 ENDCLASS.
 CLASS zcl vehicle xx IMPLEMENTATION.
   METHOD constructor.
     object count = object count + 1.
     id = object count.
   ENDMETHOD.
   METHOD show.
     DATA msg TYPE string.
     msg = `Vehicle`
                             && |{ id }| &&
            `, Speed = ` && |{ speed }| &&
`, Max-Speed = ` && |{ max_speed }|.
     MESSAGE msg TYPE 'I'.
   ENDMETHOD.
   METHOD speed up.
     speed = speed + step.
     IF speed > max speed.
        speed = max_speed.
     ENDIF.
   ENDMETHOD.
   METHOD stop.
     speed = 0.
   ENDMETHOD.
 ENDCLASS.
 Check. 🙆 Save 🖳 and Activate 🔟 the class (acknowledge all entries of the activation pop-up ).
                              🔡 Form-Based
8. Switch back to Form-Based mode
                                             and play around in that mode by double clicking the class
```



components.

다 Class Edit Goto Utilities Environment	System Help	SAP
	④ 😪 🕒 🌐 🗟 🏝 🕰 🏝 🏖 🖼 📰 🖉 🕼	
Class Builder: Change Class	ZCL_VEHICLE_XX	
✦✦》????	晶 🚊 🗉 🍞 🚹 🛛 🔄 Local Definitions/Implementations 🗋 🗋 Class constructor 📔 🕃 Source Code-Based	Class documentatic
MIME Repository	Class Interface ZCL_VEHICLE_XX Implemented / Active	
La Repository Browser	Properties Interfaces Friends Attributes Methods Events Types Aliases	
निः Repository Information System		
Tag Browser	🕒 Parameter 🕴 Exception 🖹 🛛 🛱 🚭 🔂 🔛 🔜 💥 📬 🕮 🐣 🕼 🔀 🧭 💭 Filter	
📮 Transport Organizer	Method Level Visi M. Description	
Test Repository	CONSTRUCTOR InstancPubli 🕞	
	SPEED_UP InstancPubli	
Local Objects 🔳	STOP InstancPubli	_
ABAP_00_XX 🗸 😪	SHUW InstandPubli	_
Object Name I		
☑ \$TMP ABAP_00_XX		
🗢 🗋 Class Library		
D ZCL_VEHICLE_XX V		
	▷ M30 (1) 800	1022 INS 🖌

Tip: Select Signature when displaying the implementation of a method.

	弖	=]		😭 💽 Patte	rn Pretty Printer	Signature		
	Ty	.	Para	ame	ter	Type spec.	Descri		
	Þ		STE	P		TYPE I			
	Meti	hoo	4		SPEED_UP				
			1	Ξ	method sp	peed_up.			
			2		speed =	= speed + step).		
			3	白	if spea	ed > max_speed	l.		
		4 speed = max_speed.							
			5	ŀ	endif.				
9.			6	L	endmethod	1.			

Create an application class

Create a class ZCL_APPLICATION_XX (where XX is your group number).

- This class should use your vehicle class.
- It should have simply one static method START that creates and uses objects of ZCL_VEHICLE_XX.

Solution

1. Create the class in the object navigator, where you can directly right click Classes now



Local Objects 🛅	
ABAP_OO_XX 🗸 🗸	
	X
Object Name	D
🗢 🗋 Class Library	
Create	_
	eh

2. Fill the pop-up as follows (where XX is your group number).

🖻 Create Class					
Class	ZCL_APPLICATION_XX				
Description	Application class				
Instantiation	Abstract 🗈				
Class Type					
Osual ABAP Cla	ISS				
O Exception Class	\$				
With Message Class					
O Persistent class					
O Test Class (ABAP Unit)					
🗹 Final					
🖌 Save 🔀					

3. Implement the class as follows:

```
CLASS zcl_application_xx DEFINITION PUBLIC ABSTRACT FINAL CREATE PUBLIC.
  PUBLIC SECTION.
    CLASS-METHODS start.
  PROTECTED SECTION.
  PRIVATE SECTION.
ENDCLASS.
CLASS ZCL APPLICATION XX IMPLEMENTATION.
  METHOD start.
    DATA vehicle
                    TYPE REF TO zcl vehicle xx.
   DATA vehicle tab LIKE TABLE OF vehicle.
   DATA tabindex TYPE i.
    DO 10 TIMES.
     CREATE OBJECT vehicle.
     APPEND vehicle TO vehicle tab.
   ENDDO.
    LOOP AT vehicle tab INTO vehicle.
      tabindex = sy-tabix * 10.
      vehicle->speed_up( tabindex ).
      vehicle->show( ).
    ENDLOOP.
  ENDMETHOD.
ENDCLASS.
```



Check, 🚰 Save 🖳 and Activate 🔟 the class (acknowledge all entries of the activation pop-up).

4. Select 🖼 (F8) in the Class Builder and execute method START.



Now you can see the lds and speed of the objects created.

C Information			
Yehicle 8, Speed = 50, Max-Speed = 50			

- 5. To examine further, navigate to the source code of method START and create a breakpoint at an appropriate position. To add a breakpoint at a certain line, double-click in the left margin on the line that you want a breakpoint. A
 - little ¹ should appear in the margin on that line. Similarly, this is the same way you remove a breakpoint.

6.		
C Method Edit Goto Utilities(M) Environmer	nt System Help	
	😧 🞗 💄 🖁 🎼 谷 谷 ଣ 🕱 🔟 🖗 🖷	
Class Builder: Class ZCL_APF	PLICATION_XX Change	
	🔓 🖾 🚺 🛛 🝘 🚳 Pattern Pretty Printer Signature 📔 Public Section	🔄 Protecter
MIME Repository	Ty. Parameter Type spec. Descri	
🖧 Repository Browser		
張Repository Information System		
Tag Browser	Method START	Active
🖶 Transport Organizer	1 🖂 METHOD start.	
Test Repository	2 DATA VENICLE TYPE REF TO ZCI_VENICLE_XX.	
	4 DATA tabindex TYPE i.	
Local Objects	5 DO 10 TIMES.	
ABAP OO XX	6 CREATE OBJECT vehicle.	
	APPEND vehicle TO vehicle_tab.	
	8 - ENDDO.	
Object Name	9 🗗 LOOP AT vehicle_tab INTO vehicle.	
	10 tabindex = sy-tabix * 10.	
	11 vehicle->speed_up(tabindex).	
	Scope \METHOD start\DO ABAP Ln	6 Col 29
		• •
	▷ M30 (1) 800 🖳 iwdfvm3022 II	18 🔰 🎢

Test the method again and play around with the ABAP Debugger.



Exercise 2, Inheritance (Solution in Package Z_ABAP_OBJECTS_INTRODUCTION_B)

Create a truck subclass

Create a subclass ZCL_TRUCK_XX (where XX is your group number).

The class should have an instance constructor that sets the maximal speed to 100

It should redefine the method SHOW to produce some specific output.

Solution

1. Create the class in the object navigator as before, but select **Create inheritance** () this time in order to enter a **Superclass**.

🖻 Create Class	×					
Class	ZCL_TRUCK_XX					
Superclass	ZCL_VEHICLE_XX					
Description	Truck class					
Instantiation	Public 🛅					
Class Type						
Usual ABAP Cla	ISS					
O Exception Class						
□ With Message Class						
O Persistent class						
O Test Class (ABA	AP Unit)					
🗹 Final						
🖌 Save 🗶						



2. Enter the truck's own constructor in the **Form-based** mode (type it or select **Create constructor** double click it and implement it as follows:

Constructor	١
_	1





3. Select method SHOW in the **Form-based** mode and redefine it by selecting **(3)**. Replace the implementation as follows:



Create a ship subclass

Create a subclass ZCL SHIP XX (where XX is your group number).

Check out the syntax in the Source code-based mode.

- The class should have an instance constructor that sets the maximal speed to 30 and that has an import parameter to set an additional read-only attribute to the ship's name.
- It should redefine the method SHOW to produce some specific output (including the ship's name).

Solution

- 1. Create the subclass ZCL_SHIP_XX in the object navigator as you did before for the truck class (where XX is your group number)..
- 2. Add a new public instance attribute NAME of type STRING either in **Form-based** mode or in **Source code-based** mode.

Class Interface	ZCL_SHIP_99		Impleme	nted / Inactive	(revised)		_	
Properties Inter	faces Friends	Attributes	Methods	Events	Types Aliases			
Attribute	Level	Visibility	Read-Only	Typing	Associated Type		Description	Initial value
<zcl_vehicle_99></zcl_vehicle_99>								
SPEED	Instance Attribute	Protected		Туре	I	\$		
MAX_SPEED	Instance Attribute	Protected		Туре	Ι	\$		50
NAME	Instance Attribute	Public		Туре	STRING	6 \$		
				Tyne		b		

3. Insert the ship's own constructor as you did for the truck class.

4. Create an importing parameter for the constructor either by selecting Parameter in Form-based mode

Class In	iterface		ZCL_SI	HIP_XX			Impler	mented
Pro	operties	Inter	faces	Friends	Attribu	ites Met	hods	E
Methoo	l paramet	ters	CI	ONSTRUCTOR				
🗢 м	ethods	🖏 Exc	eptions		3 🖬 💧	(🖻 🛱		
Param	Pass	О., Ту	oing M	Associated T	уре	Default valu	je D)escrip [.]
NAME		Ту	pe	STRING				

or by adding it in Source code-based mode:

```
METHODS constructor
IMPORTING
name TYPE string.
```



5. Implement the constructor as follows:

```
METHOD CONSTRUCTOR.
  super->constructor().
  max_speed = 30.
  me->name = name.
ENDMETHOD.
```



6. Redefine method SHOW as follows:

```
METHOD show.
DATA msg TYPE string.
msg = me->name &&
    `, Speed = ` && |{ speed }| &&
    `, Max-Speed = ` && |{ max_speed }|.
MESSAGE msg TYPE 'I'.
ENDMETHOD.
```

7. Save 📙 and Activate 🚺

Check out the syntax in the Source code-based mode.

Adjust the application class

The code of the START method should demonstrate the usage of the subclasses now.

- Declare extra reference variables TRUCK and SHIP for the new classes.
- You can delete the code that creates objects for VEHICLE. Instead, create one instance of each of your new subclasses and place the corresponding reference into VEHICLE_TAB.
- Call the method SPEED_UP for both classes using the respective subclass reference, and SHOW using a superclass reference.

Solution

1. Replace the code of method START of ZCL_APPLICATION_XX with the following (where XX is your group number):

Note the polymorphic method call vehicle->show().

2. Execute method START from the Class Builder again.



Exercise 3, Interfaces (Solution in Package Z_ABAP_OBJECTS_INTRODUCTION_C)

Create a status interface

- Create an interface ZIF_STATUS_XX (where XX is your group number).
- The interface should have one instance method SHOW.

Solution

1. Create the interface in the object navigator as follows:



Interface	ZIF_STATUS_XX]				
Description	Status Interface]				
🗌 Only Modeled	Only Modeled					
Save 🗶						

2. Define one Method without parameters:







Implement the interface in the superclass

Implement ZIF_STATUS_XX in ZCL_VEHICLE_XX (where XX is your group number).

- Copy the implementation of the class method SHOW to the interface method SHOW.
- Delete class method SHOW
- Create an alias name SHOW for the interface method in order to keep the subclasses valid.

Solution

1. Open ZCL_VEHICLE_XX and enter the interface either in Form-based mode:

_	Class Interface	ZCL_V	EHICL	E_XX			Im	oleme	ented / Active
	Properties Int	erfaces	Frie	nds	ł	Attributes	Metho	ds	Events
	Interface			Abstra	ct	Final	Modele	Desc	ription
	ZIF_STATUS_XX								

or in Source code-based mode:

PUBLIC SECTION.
 INTERFACES zif_status_xx.

2. Implement the interface method with the code from SHOW:

```
METHOD zif_status_xx~show.
DATA msg TYPE string.
msg = `Vehicle` && |{ id }| &&
`, Speed =` && |{ speed }| &&
`, Max-Speed =` && |{ max_speed }|.
MESSAGE msg TYPE 'I'.
ENDMETHOD.
```

- 3. Delete (
- 4. Create a public alias SHOW:

CLE_XX Implemented / Active	ZCL_VEHICLE_XX	Class Interface
Friends Attributes Methods Events Types Aliases	Interfaces Friends	Properties
Filter		
Ty. Alias Visible	nt	Interface Compor
😵 SHOW Public	IOW	ZIF_STATUS_XX~
Ty. Alias Visible	ntenaces prinends nt IOW	Interface Compor ZIF_STATUS_XX~

5. Save 🖳 and Activate 🚺..

6. Check ZCL_TRUCK_XX and ZCL_SHIP_XX (where XX is your group number). They redefine the superclass interface method SHOW via the alias SHOW now and there should be no errors.



Create a new helicopter class that also implements the interface

Create a new class ZCL_HELICOPTER_XX (where XX is your group number) that is not part of the ZCL_VEHICLE_XX inheritance tree but that also implements STATUS.

Solution

1. Create class ZCL_HELICOPTER_XX (where XX is your group number) as follows:

```
CLASS zcl_helicopter_xx DEFINITION PUBLIC FINAL CREATE PUBLIC .
   PUBLIC SECTION.
   INTERFACES zif_status_xx.
   PROTECTED SECTION.
   PRIVATE SECTION.
ENDCLASS.
CLASS zcl_helicopter_xx IMPLEMENTATION.
   METHOD zif_status_xx~show.
   DATA msg TYPE string.
   msg = `Helicopter, idle`.
   MESSAGE msg TYPE 'I'.
   ENDMETHOD.
ENDCLASS.
```

2. Save 🖳 and Activate 🚺

Adjust the application class

The code of the START method should demonstrate the usage of the interface now.

- Declare a reference variable HELI for the class ZCL_HELICOPTER_XX following (where XX is your group number) and create a corresponding object.
- Replace the reference variable VEHICLE and the table VEHICLE_TAB with an interface reference STATUS and an internal table STATUS_TAB.
- Insert the reference variables for truck, ship and helicopter into the table STATUS_TAB.

Solution

1. Replace the code of method START of ZCL_APPLICATION_XX with the following (where XX is your group number):

```
METHOD start.
                   TYPE REF TO zif status xx,
  DATA: status
         status tab LIKE TABLE OF status,
         truck TYPE REF TO zcl truck xx,
        ship TYPE REF TO zcl_ship_xx,
heli TYPE REF TO zcl_helicopter_xx.
                TYPE REF TO zcl ship xx,
  CREATE OBJECT: truck,
                   ship EXPORTING name = 'Titanic',
                   heli.
  APPEND: truck TO status tab,
          ship TO status_tab,
heli TO status_tab.
  truck->speed up( 30 ).
  ship->speed_up( 10 ).
  LOOP AT status_tab INTO status.
    status->show( ).
  ENDLOOP.
ENDMETHOD.
```

Note the polymorphic method call status->show().

2. Execute method START from the Class Builder again.



Exercise 4, Events (Solution in Package Z_ABAP_OBJECTS_INTRODUCTION_D)

Define an event in the ship class

Objects of ZCL_SHIP_XX (where XX is your group number) should raise an event if the speed becomes higher than the maximal speed.

Define an instance event DAMAGED in ZCL_SHIP_XX

Raise the event in method SPEED_UP.

Solution

1. Open ZCL_SHIP_XX (where XX is your group number) in the Class Builder and define an instance event DAMAGED either in **Form-based** mode:

Class Interface ZCL_	SHIP_XX	Implemented / Active				
Properties Interfaces	Friends Attributes	Methods Events	Types Aliases			
→ Parameters DE C E E X C E A A A A C Filter						
Event	Туре	Visibility				
DAMAGED	Instance Event	Public				

or in Source code-based mode:

PUBLIC SECTION.

2. Redefine (1) method SPEED_UP and implement it as follows:

```
METHOD SPEED_UP.
speed = speed + step.
IF speed > max_speed.
max_speed = 0.
CALL METHOD stop.
RAISE EVENT damaged.
ENDIF.
ENDMETHOD.
```

3. Save 📙 and Activate 🚺.

Define an event handler method in the helicopter class

The helicopter class should be able to handle the event DAMAGED of the ship class.

- Define a public method RECEIVE as an event handler for event DAMAGED of ZCL_SHIP_XX in
- ZCL_HELICOPTER_XX (where XX is your group number).
- Implement the event handler in a way that is simply sends a message that uses the default event parameter SENDER to address the name of the damaged ship.



Solution

1. Open ZCL_HELICOPTER_XX (where XX is your group number) in the class builder.

In **Form-based** mode, enter a new public method RECEIVE and select **Detail view** (**D**) where you can define the method as an **Event handler**:

Class Interface	ZCL_HELICOPTER_XX				Implemented / Active			
Properties	Interfaces Frie	Methods	Event	ts Tyr				
Parameter [Crange Method RECEIVE							
Method	Object type	ZCL_HE	LICOPTER_XX					
ZIF_STATUS_XX~	Method	RECEIV	E					
RECEIVE	Description							
	Visibility		Method					
	O Public		🔿 Static					
	O Protected		Instance	e				
	Private							
	Abstract							
	🗌 Final							
	Event handler for							
	Class/interface	ZCL_SH	IP_XX					
	Event	DAMAGE	D					
	Modeled							
	Editor lock							
	Active							
	Created by							
	Last changed by (Defn)							
	Last changed by (Imp.)							
	Change 🔀							

and define the predefined importing parameter SENDER:

Method parameters	RECEIVE	E					
🗢 Methods 🕅 Exception	s 🔳 [3	36	1 🗶 🖻			
Parameter	Туре	Pa_	0	Typing M	Associated Type	Default value	Description
SENDER	Importin	۱ 🗖					>SENDER<

In **Source code-based** mode it is enough to type:

METHODS receive FOR EVENT damaged OF zcl_ship_xx. IMPORTING sender.



2. Implement method RECEIVE as follows:

```
METHOD receive.
DATA msg TYPE string.
msg = `Helicopter received call from ` && sender->name.
MESSAGE msg TYPE 'I'.
ENDMETHOD.
```

3. Save 🖳 and Activate 🚺.

Adjust the application class

The code of the START method should register the instances of the helicopter class for the events of the ship class.

- Use statement SET HANDLER to register the event handler.
- Call method SPEED_UP of the ship object in order to raise the event.

Solution

1. Replace the code of method START of ZCL_APPLICATION_XX with the following (where XX is your group number):

```
METHOD start.
  DATA: status
                 TYPE REF TO zif status xx,
            status_tab LIKE TABLE OF status,
            truck TYPE REF TO zcl_truck_xx,
            ship TYPE REF TO zcl_ship_xx,
           heli
                 TYPE REF TO zcl helicopter xx.
  CREATE OBJECT: truck,
                 ship EXPORTING name = 'Titanic',
                 heli.
  APPEND: truck TO status_tab,
         ship TO status_tab,
         heli TO status_tab.
  SET HANDLER heli->receive FOR ALL INSTANCES.
  truck->speed up( 30 ).
  ship->speed_up( 10 ).
 LOOP AT status tab INTO status.
   status->show().
  ENDLOOP.
  DO 5 TIMES.
   ship->speed up( 10 ).
  ENDDO.
ENDMETHOD.
```

2. Execute method START from the Class Builder again.



Exercise 5, Exceptions (Solution in Package Z_ABAP_OBJECTS_INTRODUCTION_E)

Define an exception class

Objects of ZCL_VEHICLE_XX should raise an exception ZCX_OVERHEATING_XX (where XX is your group number) if the speed becomes higher than the maximal speed.

- Define an exception ZCX_OVERHEATING_XX (where XX is your group number)
- Raise the exception in method SPEED_UP.

Solution

- 1. Open ZCL_VEHICLE_XX (where XX is your group number) in the Class Builder, place the cursor on method SPEED_UP in **Form-based** mode and select the button **Exception**.
- 2. Mark Exception Classes and enter ZCX_OVERHEATING_XX (where XX is your group number).

Method exceptions	SPEED_UP	
🗢 Methods 🗖 Paramete	ers 🗐 🖪 🛃 🙀 🎁 📳 🗹 Exception Classes	
Exception	Resumable Description	
ZCX_OVERHEATING_XX		

3. Save 📕 and select Yes and Save on the following pop-ups with respectively.







- 4. Double click the name of exception class, Activate it and return to the vehicle class.
- 5. Change the implementation if method SPEED_UP as follows (where XX is your group number).:

```
METHOD speed_up.
speed = speed + step.
IF speed > max_speed.
RAISE EXCEPTION TYPE zcx_overheating_xx.
ENDIF.
ENDMETHOD.
```

6. Save 🖳 and Activate 🚺

Adjust the application class

The code of the START method must handle the exception now.

- Use a TRY control structure to handle the exception ZCX_OVERHEATING_XX (where XX is your group number)
- Call method SPEED_UP of the truck object in order to raise the exception.

Solution

 Check the syntax of method START of ZCL_APPLICATION_XX (where XX is your group number). You should get the following warnings:

Description	Row	Туре
Class ZCL_APPLICATION_XX,Method START	20	040
The exception ZCX_OVERHEATING_XX is neither caught nor is it declared		
in the RAISING clause of "START".		
Class ZCL_APPLICATION_XX,Method START	15	040
The exception ZCX_OVERHEATING_XX is neither caught nor is it declared		
in the RAISING clause of "START".		
Class ZCL_APPLICATION_XX,Method START	14	040
The exception ZCX_OVERHEATING_XX is neither caught nor is it declared		
in the RAISING clause of "START".		

2. Adjust the implementation of method START as follows (where XX is your group number):

```
METHOD start.
 DATA: status
                 TYPE REF TO zif status xx,
       status tab LIKE TABLE OF status,
       truck TYPE REF TO zcl truck xx,
       ship TYPE REF TO zcl ship xx,
       heli TYPE REF TO zcl helicopter xx.
 CREATE OBJECT: truck,
                 ship EXPORTING name = 'Titanic',
                 heli.
 APPEND: truck TO status tab,
         ship TO status tab,
         heli TO status tab.
  SET HANDLER heli->receive FOR ALL INSTANCES.
 TRY.
     truck->speed up(30).
     ship->speed up(10).
     LOOP AT status tab INTO status.
       status->show( ).
     ENDLOOP.
     DO 5 TIMES.
```



```
ship->speed_up(10).
ENDDO.
DO 5 TIMES.
truck->speed_up(30).
truck->show().
ENDDO.
CATCH zcx_overheating_xx.
MESSAGE 'Truck overheated' TYPE 'I'.
EXIT.
ENDTRY.
ENDMETHOD.
```

No syntax warnings should occur any more.

3. Execute method START from the Class Builder again.

Exercise 6, Unit Tests

(Solution in Package Z_ABAP_OBJECTS_INTRODUCTION_F)

Define a test class with a test method for the vehicle class

Class ZCL_VEHICLE_XX (where XX is your group number) should contain a test class that tests the SPEED_UP method completely.

Declare and implement a local test class TEST_VEHICLE in the vehicle class.

Solution

1. Open ZCL_VEHICLE_XX (where XX is your group number) in the Class Builder an select the following:

⊡ Classdit	<u>G</u> oto <u>U</u> tilities En <u>v</u> ironment System	Help)	
©	Local Definitions/Implementations	•	Local Definitions/Implementati	ons N
Class Buil	<u>S</u> ections	F	<u>Class relevant local Definitions</u>	, 45 -
	<u>T</u> ext elements		Local <u>T</u> est Classes	_
← 🖻 🕅 😢 😢	Persistence representatn		<u>M</u> acros	IS
ABAP Unit Brov	Proxy Editor		Class Interface ZCL_	VEHICLE_XX
MIME Reposito	<u>D</u> ocumentation	•	Properties Interfaces	Friends
🖁 Repository Bro	Tr <u>a</u> nslation			
निः Repository Info	Object directory entry		Parameter Exception	1 🖆 🖻 🖬
ETag Browser	Applicatio <u>n</u> help		Method	Level Visi
🖶 Transport Orga	<u>B</u> ack		ZIF_STATUS_XX~SHOW	InstancPubl

2. Enter the following code there (where XX is your group number):

CLASS test_vehicle DEFINITION DEFERRED. CLASS zcl_vehicle_xx DEFINITION LOCAL FRIENDS test_vehicle.

The second statement is the important one. It declares the test class as friend of the global class in order to have access of the vehicle's private components. The first statement is necessary for the ABAP Compiler. (select F1 to learn more).



3. Navigate back to the Class Builder's main screen and select Local Test Classes in the above menu, answer the pop-up with Yes, and implement your test class as follows (where XX is your group number):

```
CLASS test vehicle DEFINITION FOR TESTING
                   RISK LEVEL HARMLESS
                   DURATION SHORT.
  PRIVATE SECTION.
    DATA vehicle TYPE REF TO zcl vehicle xx.
    METHODS: test speed up FOR TESTING,
            setup,
            teardown.
ENDCLASS.
CLASS test vehicle IMPLEMENTATION.
  METHOD setup.
   CREATE OBJECT vehicle.
  ENDMETHOD.
  METHOD test_speed_up.
    TRY.
        vehicle->speed up( 50 ).
        cl_abap_unit_assert=>assert_equals(
          EXPORTING
           exp = 50
            act = vehicle->speed
            msg = 'Speed not as expexted'
            level = if aunit constants=>critical ).
      CATCH zcx overheating_xx.
        cl abap unit assert=>fail(
          EXPORTING
           msg = 'No exception expected'
            level = if aunit constants=>critical ).
    ENDTRY.
    TRY.
        vehicle->speed up( 1000 ).
        cl abap unit assert=>fail(
          EXPORTING
            msg = 'Exception expected'
            level = if aunit_constants=>critical ).
      CATCH zcx overheating xx.
    ENDTRY.
  ENDMETHOD.
  METHOD teardown.
    CLEAR vehicle.
  ENDMETHOD.
ENDCLASS.
Save
```

4. Navigate back to the Class Builder's main screen, Save 🖳 and Activate 🔟.

5. Carry out the test method by

Class -> Unit Test in the Class Builder menu

or

Test -> Unit Test from the context menu of the class in the Repository Browser

The result should be:



Use the ABAP Unit Browser

Examine the possibilities of the ABAP Unit Browser embedded in the Object Navigator

Solution

 Select Utilities -> Settings in the Object Navigator and additionally select the ABAP Unit Test Browser under Workbench (General):

🕞 User-Specific Settings		×
Workbench (General) ABAP Editor Class Builder	Screen Painter	
Initial Screen]	
Browser Selection]	
Repository Browser		
Repository Information System		
🗹 Transport Organizer		
MIME Repository		
☑ Tag Browser		
✓ Test Repository		
ABAP Unit Test Browser		

2. Select the ABAP Unit Browser:

ABAP Unit Browser	
Construction of the	

3. Select Class Pool, enter ZCL_VEHICLE_XX (where XX is your group number), and carry out the test with measuring the test coverage:



Class Pool 📑		Applic, Co
ZCL_VEHICLE_XX	66	Package
		🗌 🗖 Main
중 숲 🛗 📮 🖬 🖬		Surroun
Component / Object		Allowed
▽ 🔀 \$TMP		Pack:
🗢 🥥 Classes		
		Tranana
🗢 🥥 Test Classes	<u>E</u> xpand subtree	
V 🥥 TEST_VEH	Collapse subtree	
V 🖉 Method	Find	
D 🥥 TES		
D 🏷 ABAP U_	Helb	
	Display Object	
	Object Directory Entry	
	<u>E</u> xecute	
	Execute and measure covera	ge N
		K

4. After successful execution you can navigate through the results to view the test coverage:

I⊡ <u>R</u> esult <u>E</u> dit System <u>H</u> elp	
S	2 😡 I 🗅 Hi Hi 1 2 2 4 4 1 🕱 🖉 I 🖗 🖪
ABAP Unit: Result Display	
	Alerts and Messages
Task/Program/Class/Method Stat F	Type Message
✓ C ZCL VEHICLE XX======= □	Coverage Results for Program: ZCL_VEHICLE_XX========CP
<pre>G TEST_SPEED_UP</pre>	
	Coverage for Package \$TMP of Program ZCL VEHICLE XX===================================



C F	esult	<u>E</u> dit S <u>y</u> ste	ım <u>H</u>	elp						SAP
0				1 4 📙 I 😋 🚱 🔇) I 🗅 (A) (A) I	8008) 🔀	2 🛛 🖪		
A	BAP	Unit: Re	sul	t Display						
	85	7 🛗 🖂	70	🔠 🕄 Filter 🗈						
	Progra	m Level Cov	erage l	oy Statements						
	Cov	Proc. Cov.	Stn	nnt Cov. Tcode/Program			Obj.	Object Name		· · · · · · · · · · · · · · · · · · ·
	Δ	50,00		58,33 ZCL_VEHICLE_XX	(======================================	===CP	CLAS	ZCL_VEHICLE	_XX	
[• •									• •
	<u>a</u> 7	7 🛗 🖾	78							
	Proced	lure Level Co	verage	by Statements						
B	Cov	Stmnt Cov.	Туре	Name	Class of the Pro	cessing Block		₅ Statements	Е	Invocation
		100,00	METH	CONSTRUCTOR	ZCL_VEHICLE_	×X		3		1
		100,00	METH	SPEED_UP	ZCL_VEHICLE_	××		4		2
	۲	0,00	METH	STOP 'N	ZCL_VEHICLE_	~		2		<u>م</u> ا
	<u> </u>	0,00	METH	ZIF_STATUS_XX~SHOW	ZCL_VEHICLE_	~~		3		
	▲ ►!									
	🕨 M30 (2) 800 🖪 iwdfvm3022 INS 🥖 🥢									

도 Statement Coverage Edit Goto System Help	SAP
■ Q I Q I Q I Q I Q I Q I Q I Q I Q I Q	
ABAP Unit: Statement Coverage	
Program ZCL_VEHICLE_XX	
1 Image: Method speed_up. 2 speed = speed + step. 3 If speed > max_speed. 4 RAISE EXCEPTION TYPE zcx_overheating_xx. 5 ENDIF.	
6 ENDMETHOD.	
M30 (2) 800) 🖪 iwdfvm3022 INS 🥖



Exercise 7, Service Enablement (Solution in Package ZABAP_OO_SERVICEENABLEMENT

Expose Method As Web Service (Inside-Out Approach)

In this exercise,

- You will create a RFC-enabled Function Module to invoke the IF_DEMO_CR_CAR_RENTL_SERVICE~MAKE_RESERVATION method from Class CL_DEMO_CR_CAR_RENTAL_SERVICE
- Copy existing RFC-enabled Function Module into your Function Group
- Expose the RFC-enabled Function Modules in the Function Group as a Web Service
- Configure/Test the web service using SOAManager and Web Service Navigator
- Optionally, debug using an external breakpoint

Class Interface CL_DEM0_CR_CAR_RENTAL_SERVICE Implemente	ed / Active			
Properties Interfaces Friends Attributes Methods	Events Types Aliases			
□ Parameter 🛯 Exception 👔 🖻 🖬 🚭 🛃 🔣 🗎 📇 🔺	🛗 🔀 🧭 🗆 Filt	er		
Method	Level	Visibility	Method type	Description
IF_DEMO_CR_CAR_RENTL_SERVICE~CREATE_CUSTOMER	Instance Method	Public		Create Customer
IF_DEMO_CR_CAR_RENTL_SERVICE~GET_CARS_BY_CATEGORY	Instance Method	Public		Get Rental Cars by Category
IF_DEMO_CR_CAR_RENTL_SERVICE~GET_CUSTOMER_BY_ID	Instance Method	Public		Get Customer by ID
IF_DEMO_CR_CAR_RENTL_SERVICE~GET_CUSTOMERS_BY_NAME	Instance Method	Public		Get customers by name
IF_DEMO_CR_CAR_RENTL_SERVICE~GET_RESERVATIONS_BY_CUST_ID	Instance Method	Public		Get Reservation by Customer ID
TF_DEMO_CR_CAR_RENTL_SERVICE~MAKE_RESERVATION	Instance Method	Public		Create Reservation
GET_SERVICE	Static Method	Public		Get Singleton for Service
CLASS_CONSTRUCTOR	Static Method	Public	- -	Static Constructor

Solution

- 1. Logon to the system and open the Object Navigator of the ABAP Workbench (Transaction SE80, enter /nSE80 in the command field of the system task bar).
- 2. Select Local Objects in order to work in a test package that is not transported to other systems.



Hit Enter.



- 3. Create a Function Group
 - Right Click the name of the local package and navigate to the creation of a Function Group.
 - Enter **ZFG_XX** (where XX is your group number) and short text. Click **Save**.

			🖻 Create Function Grou	p	×
≙ # ≋`€ ⊠			Function group	ZFG_XX	
Create >	Development Coordination	-	Short text	Function Group for ABAP OO - Group XX	
- Change Display	Program Eunction Group				
Copy	Web Dynpro		Person Responsible	ABAP_00_99	
Check +	Class Library				
Display Object Directory Entry Write Transport Entry	Enterprise Service BSP Library				
Where-Used List Other Functions	Enhancement		Save 🗙		
	Image Image Organization Image Display Image Operate Image Operate Image Operate Image Operate Image Operate Image Operation Image Operation Image Operation Image Other Functions Image	Image Description Create Development Coordination Change Program Display Fraction Group Copy Web Dynpro Delete Dictionary Object Check Class Library Display Object Directory Entry BSP Library Where-Used List Enhancement Other Functions Business Engineering	Image Description Create Development Coordination Change Program Display Function Group Cgpy Web Dynpro Display Class Library Display Object Directory Entry Bevelopment Coordination Write Transport Entry Bevelopment Coordination Write Transport Entry Bevelopment Coordination Other Functions Enterprise Service	Image: Construction Group Create Development Coordination Change Program Display Function group Copy Web Dynpro Display Class Library Display Object Directory Entry Breprise Service White Transport Entry Business Engineering Other Functions Constructions	Image: Construction group Cereate Description Change Program Display Enterprise Service Display Object Directory Entry Where-Used List Other Functions

• Acknowledge the following window without changes (select either Save or Local Object).

📴 Create Object Directory E	ntry	
Object R3TR FUGR ZF	G_XX	
Attributes		
Package	<u>\$TMP</u>	6
Person Responsible	ABAP 00 99	
Original System	<u>M30</u>	
Original language	EN English	
Created On		
🕻 🔚 🕽 Local Object 🛛 📈 Lo	ck Overview 🔁 🗶	

- 4. Create the function module.
 - Expand the Function Groups Folder. Right click on the function group, **ZFG_XX**, and choose Create->Function Module

Object Name			Desc	ription
🗢 🗋 \$TMP ABAP_O				
👂 🗋 Class Libra				
🗢 🗖 Function Gr				
D T ZEG XX				
	<u>Create</u>			Function <u>G</u> roup
C <u>h</u> ange				Eunction Module
	<u>D</u> isplay		•	S <u>u</u> broutine

• Enter zCustomerReserveCar_XX (where XX is your group number) and short text. Click Save.



• You can disregard the following informational message whenever it may appear





- 5. Define the function module interface by entering its parameters
 - Provide the import parameters for the function module under **Import** tab

	Parameter Name	Typing	Associated Type	Pass Value
	CUSTOMER_ID	ТҮРЕ	DEMO_CR_CUSTOMER_ID	Ŋ
	CATEGORY	ТҮРЕ	DEMO_CR_CATEGORY	K
	STARTDATE	ТҮРЕ	DEMO_CR_DATE_FROM	\checkmark
ĺ	ENDDATE	ТҮРЕ	DEMO_CR_DATE_TO	\checkmark

Function module	Inactive	(Revised)					
Attributes Import Export Changing Tables				eptions S	ource code		
Parameter Name	Typing	Associated Type		Default value	Optional	Pass Value	Short text
CUSTOMER_ID	ТҮРЕ	DEMO_CR_CUSTOMER_ID				V	Customer ID
CATEGORY	ТҮРЕ	DEMO_CR_CATEGORY				v	Car Category
STARTDATE	TYPE	DEMO_CR_DATE_FROM				V	Start Date
ENDDATE	TYPE	DEMO_CR_DATE_TO				Image: A state of the state	End Date

• Provide the export parameters for the function module under Export tab

			Pass
Parameter Name	Typing	Associated Type	Value
RESERVATIONS	TYPE	DEMO_CR_RESERVATIONS_TT	У
RETURN	TYPE	BAPI_MSG	Z

Function module	ZCUSTOMERRE	ESERVECAR_XX Active		
Attributes Import	Export	Changing Tables E	xceptions	Source code
Parameter Name	Typing	Associated Type	Pass Val	Short text
RESERVATIONS	ТҮРЕ	DEMO_CR_RESERVATIONS_TT		Table Type for Reservation
RETURN	ТҮРЕ	BAPI_MSG		Message Text

• Please mark "Remote-enabled Module" radio button under Attributes tab.

Function module	CUSTOMERRESERVECAR_XX Inactive (Re	evised)	
Attributes Import	Export Changing Tables Except	ions Source code	
Oleveifferhing			
Function Group	ZE6 XX Eunction Grou	In for ABAP OO - Group XX	
Short Text	Reserve Car for Customer		
Processing Type		General Data	
O Normal Function Modul	e	Person Responsible	ABAP_00_99
Remote-Enabled Modu	le 📃 🗖 BasXML supported	Last Changed By	ABAP_00_99
O Update Module		Changed on	13.09.2009
Start immed.		Package	\$TMP
O Immediate Start, No	Restart	Program Name	SAPLZFG_XX
O Start Delayed		INCLUDE Name	LZFG_XXU01
O Coll.run		Original Language	EN
		Not released	
		Edit Lock	
		Global	



6. Complete the source code for your function module under Source Code tab.

Function module ZCUSTOMERRESERVECAR_XX Inactive	
Attributes Import Export Changing Tables Exceptions Source code	
	You can toggle between Display and Change mode
1 E FUNCTION zcustomerreservecar_xx .	
2	
3 *"*"Local Interface:	using the 🖾 icon.
4 *" IMPORTING	
5 *" VALUE(CUSTOMER_ID) TYPE DEMO_CR_CUSTOMER_ID	
6 *" VALUE (CATEGORY) TYPE DEMO_CR_CATEGORY	
7 *" VALUE (STARTDATE) TYPE DEMO_CR_DATE_FROM	
8 *" VALUE (ENDDATE) TYPE DEMO_CR_DATE_TO	
9 *" EXPORTING	
10 *" VALUE (RESERVATIONS) TYPE DEMO_CR_RESERVATIONS_TI	
11 *" VALUE (RETURN) TYPE BAPI_MSG	
12 *"	
13) Insert code here	
14	
15 ENDFUNCTION .	

Add the ABAP source code

```
DATA: lr service TYPE REF TO if_demo_cr_car_rentl_service,
       lo exception TYPE REF TO cx root,
       l customer TYPE demo cr scustomer,
       l reservation TYPE demo cr sreservation.
  lr_service = cl_demo_cr_car_rental_service=>get_service( ).
 TRY.
     lr service->make reservation(
         EXPORTING
           i customer id = customer id
            i_category = category
            i date from = startdate
            i date to = enddate ).
   CATCH cx demo cr no customer cx demo cr lock cx demo cr reservation INTO lo excep
tion.
     return = lo exception->get text( ).
     EXIT.
    CATCH cx root INTO lo exception.
     return = lo exception->get text( ).
     EXIT.
 ENDTRY.
 l_customer = lr_service->get_customer_by_id( customer_id ).
* Method returns ALL reservations for this customer
 reservations = lr service->get reservations by cust id( customer id ).
* Delete the records not matching the requested start and end dates
 DELETE reservations WHERE date from <> startdate OR date to <> enddate.
* We still may have the situation where the same reservation request was booked multipl
e times
  CASE lines ( reservations ).
    WHEN 0.
     return = `Unable to confirm reservation - Contact Help Desk`.
   WHEN 1.
     READ TABLE reservations INTO 1 reservation INDEX 1.
     return = `Reservation ` && l reservation-
reservation id && ` Booked for ` && l customer-name.
   WHEN OTHERS.
     sort reservations by reservation id ascending.
     return = `Multipe reservation exist for ` && l customer-
name && ` in the selected time period`.
 ENDCASE.
```

Check, 🕶 Save 🖳 and Activate 🏦 the Function Module



- 7. Copy 3 additional Function Modules from a different package into your Function Group
 - Navigate to Package ZABAP_OO_SERVICEENABLEMENT click Display

Local Objects Application Hierarchy Package Program Function Group Class / Interface Internet Service Web Dynpro Comp. / Intf. BSP Application BSP Extension Inactive Objects		Description Function Gro	Package Image: Constraint of the second) Description Service Enat
Local Objects Includes	dules			

• Expand the Function Groups folder to display the Function Modules

Object Name	Description
ZABAP_00_SERVICEENABLEMENT	Service Enablement of ABAP OO Exercises
🗢 🗖 Function Groups	
	TestFMs
🗢 🔁 Function Modules	
ZCUSTOMERCREATE_XX	Create Customer
ZCUSTOMERFINDBYNAME_XX	Find Customer by Name
ZCUSTOMERGETRESERVATIONS_XX	Get List of Customer Reservations
ZCUSTOMERRESERVECAR_XX	Reserve Car for Customer
🕨 🗀 Includes	

• Right-click on the ZCUSTOMERCREATE_XX Function Module and Select Copy

Object Name		Description		
ZABAP_OO_SERVICEENABLEMENT		Service Enablement of ABAP OO Exercises		
🗢 🗋 Function Groups				
🗢 🗋 ZAOOXX		Test FMs		
🗢 🔁 Function Modules				
ZCUSTOMERCREATE_XX	-	A		1
ZCUSTOMERFINDBYNAME_XX	<u>_</u> r	reate	×	e
ZCUSTOMERGETRESERVATIONS_XX	Cł	nange		servations
ZCUSTOMERRESERVECAR_XX	Di	- snlav		her
Includes		opiaj	-	
	Cł	nec <u>k</u>		
	Ac	t <u>i</u> vate		
	Ţe	st	×	
	Co	2ру		
	<u>R</u> 6	ename		



- Enter the Following
 - To Function Module **ZCUSTOMERCREATE_99** (replace 99 with your group number)
 - Function Group **ZFG_99** (replace 99 with your group number)
 - Select Copy

📴 Copy Function Module		×
fr. Function module To Function module	ZCUSTOMERCREATE_XX ZCUSTOMERCREATE_99	
Function group	ZF6_99	
Copy 🗶		

- You can ignore any pop-up informational messages
- Repeat the same steps above to copy 2 more Function Modules remembering to replace 99 with your group number

Copy From	Сору То
ZCUSTOMERFINDBYNAME_XX	ZCUSTOMERFINDBYNAME_99
ZCUSTOMERGETRESERVATIONS_XX	ZCUSTOMERGETRESERVATIONS_99

• Select Local Objects from the dropdown box

Local Objects 🔰 🖺
ABAP_00_99 🗸 😪
Object Name
▽ 🔁 \$TMP ABAP_OO_99
👂 🛄 Class Library
👂 🛄 Function Groups

• Expand the Function Groups and Function Modules to Display your copied objects

Local Objects	
Object Name	Description
Class Library	
🗢 🗋 Function Groups	
▽ 🗋 ZFG_99	Function Group for ABAP OO - Group 99
🗢 🔂 Function Modules	
ZCUSTOMERCREATE_99	Create Customer
ZCUSTOMERFINDBYNAME_99	Find Customer by Name
ZCUSTOMERGETRESERVATIONS_99	Get List of Customer Reservations
ZCUSTOMERRESERVECAR_99	Reserve Car for Customer
🕨 🗀 Includes	



• Double-click on **ZCUSTOMERCREATE_XX** (where XX is your group number) and then click

unction Builder: Display ZCUSTOMERCRE	EATE_99
: » : • • • • • • • • • • • • • • • •	Pattern Pretty Printer Function Module Documentation
MIME Repository	Function module ZCUSTOMERCREATE 99 Inactive
Repository Browser	Attributes / Import / Export / Changing / Tables / Exceptions / Source code
Repository Information System	Tambado Import Export onanging Tableo Exceptions contrologi
Tag Browser	1 E function zoustomercreate 99.
Transport Organizer	2
Test Renository	3 *"*"Local Interface:
Enternrise Services Browser	4 *" IMPORTING
	6 *" EXPORTING
Incal Objects	7 *" VALUE (RETURN) TYPE BAPI MSG
	8 * *
	9
•10 ⇒10 V ☆ X 🗷 🛐 🛛	10 11 deta in service tune ref to if demo or cer rentl servic
bject Name Descriptio	n 12 data lo exception type ref to cx demo cr customer.
5 STMP ABAP_00_99	13 lr_service = cl_demo_cr_car_rental_service=>get_service
Class Library	14 P try.
V L Function Groups	15 Ir_service->create_customer(name).
V ZFG_99 Function G	17 o catch cx demo cr customer into lo exception.
Double ZCUSTOMERCREATE 99 Create Cu	stomer 18 return = lo_exception->get_text().
Click ZCUSTOMERFINDBYNAME_99 Find Custo	omer by Name 19 exit.
ZCUSTOMERGETRESERVATIONS_99 Get List of	Customer Reservations 20
2CUSTOMERRESERVECAR_99 Reserve C	arior Customer 21 endtry.
	22 23
	endfunction.
elect the entire worklist a	and click continue
elect the entire worklist a Plactw Objects for ABAP_00_99 Transportable Objects Local objects Object name	ind click continue
elect the entire worklist a	ind click continue
Construction of the second sec	Ind click continue
Clear the entire worklist a Charles Objects for ABAP_00_98 Transportable Objects Cocal objects Object (Objects Cocal objects Object (Objects Cocal objects D (Object (Objects Cocal objects D (Object (Objects Cocal objects D (Object (Objects Cocal objects) D (Object (Objects) D (Objects)	ind click continue
Celect the entire worklist a Constructive Objects for ABAP_00_09 Transportable Objects Cocal Objects Object raame D Object Obj name FUNC ZCUSTOMERFINGERMARE_99 FUNC ZCUSTOMERFINGERMARE_99 FUNC ZCUSTOMERFINGERMARE_99 FUNC ZCUSTOMERFINGERMARE_99 FUNC ZCUSTOMERFINGERMARE_99	and click continue
elect the entire worklist a Practive Objects for ABAP_00_09 Transportable Objects Local objects Object name D Object Objects Local objects Prunc 2005 TOMERER ATE 99 Prunc 2005 TOMERER ATE 99 Prunc 2005 TOMERER FIRESERVATIONS_99	and click continue
Belect the entire worklist a Stractwork Digets Construction Construction Diget name Diget Objects D Object Objects Construction D Object Iname Diget Objects D Object Iname Diget Objects D Object Iname Diget Objects P UNC 2005TORERCEAFE_98 Diget Objects FUNC 2005TORERCEAFE 98	Ind click continue
Celect the entire worklist a classes	and click continue
Celect the entire worklist a below the entire worklist below the entire worklist below the entire worklist below the entire worklist below the entire worklist below the entire entite enti	and click continue
Celect the entire worklist a Collect objects for ABAP_00_09 Transportable Objects Cocal objects Object raame D Object Objects Cocal objects Object raame D Object Objects Cocal objects PUNC 2CUSTOMERCRAFE_99 PUNC 2CUSTOMERCRAFE_99 PUNC 2CUSTOMERCRAFE_99 PUNC 2CUSTOMERCRAFE_99 PUNC 2CUSTOMERCRAFE_99	and click continue
Celect the entire worklist a Practive Objects for ABAP_00_99 Transportable Objects Local objects Object name D Object Objects Local objects Object name PUNO 22003TONERERATE 99 PUNO 22003TONERERATE 99 PUNO 22003TONERERATE 99 PUNO 22003TONERERATE 99 PUNO 22003TONERERATE 99 PUNO 22003TONERERATE 99 PUNO 22003TONERERATE 99	and click continue
Celect the entire worklist a Select the entire worklist a Tanspotable Objects Cost	and click continue
Celect the entire worklist a Celect the entire worklist Collect of ABAP_00_09 Transpotable Objects Construction Dotypet (objects Construction Dotypet (objects Construction Dotypet (objects Construction Constructi	and click continue
Celect the entire worklist a Construction C	and click continue
Celect the entire worklist a classical and a contract of the	and click continue
Celect the entire worklist a Celect the entire worklist Celect the entire	and click continue
Celect the entire worklist a Celect the entire worklist Collect Construction Collect Construction Collect Construction Collect Construction Collect Construction Collect Coll	and click continue

- 8. Using the Web Service Wizard, generate an Enterprise Service Definition for the function group ZFG_XX. (where XX is your Group Number).
 - Right mouse click on the function group, **ZFG_XX**, and choose Create->Other Objects->Enterprise Service





•

 On the Provide Service Definition Details step, input the name of your service definition – ZCustomerCarRental_XX(where XX is your Group Number). You can also enter a short description (Kurzbeschreibung) and set the Endpoint Type to Function Group. Click Continue

🖻 Provide Service Definition details			⊠/
Object Type Choose Endpoint Choose Operations Configure Service	Enter a name and a sho an end-point type. To change the Web Sen SE80).	rt description for the Web Service and choose vice, use the ABAP Workbench (transaction	
 Enter Package/Request Complete 	Service Definition Kurzbeschreibung Endpoint Type	ZCustomerCarRental_99 Customer Car Rental Services Function Group	
	Ba Ba	ack 🛃 Continue 🗶 Cance	1

• On the Choose Endpoint step, you have to specify the name of the function group which will serve as the implementation for this service definition. If you want to use Name Mapping (underscores are removed and changed to camel case) you can **check the Mapping of names** option.

	If you select the checkbox 'Name Mapping', the existing descriptions of the end point are applied. Initial letters are capitalized and underscores removed.
Complete	Function group Function Group ZF6_99 Mapping of names Mapping of names



• On the *Choose Operations* step, you can select which operations you wish to expose. Click Continue

Object Type Choose Endpoint Choose Operations Configure Service	Select at least one function module.	
 Enter Package/Request Complete 	Function ZCUSTOMERCREATE_99 ZCUSTOMERFINDBYNAME_99 ZCUSTOMERGETRESERVECAR_99 ZCUSTOMERRESERVECAR_99	Function Group ZF6_99 ZF6_99 ZF6_99 ZF6_99 (()) () () () () () ()) () ()) ()) () () ()) ()) () ()) ()) ()) ()) ()) ()) ()) ()) ())) ())) ())) ())) ())))) ()))))))))))))

• On the *Configure Service* step, set the **PRF_DT_IF_SEC_LOW** profile in order to set the lowest security level for this service definition. Be sure to check *Deploy Service*. If you forget to check this box, you can complete this step later, manually from transaction code **SOAMANAGER** by creating an Endpoint.

🔄 Configure Service	×	
Object Type Choose Endpoint Choose Operations Choose Operations Configure Service	Choose a profile for Security Settings. Remember that the service does not have any runtime configuration and therefore cannot be used. Create the Web service configuration in the NetWeaver Admninistrator (transaction WSADMIN2).	
 Enter Package/Request Complete 	SOAP Application SOAP Application Soap Application Profile PRF_DT_IF_SEC_LOW Authentication with User and Password, No Transport Guarantee	
	🛃 Back 🛃 Continue 🔀 Cancel]



• On the *Enter Package/Request* step, please check *Local Object* to save the generated Service Definition as local/private.

🔄 Enter Package/Request		×/
Object Type Choose Endpoint Choose Operations Configure Service Enter Package/Request	Enter the name of the package and transport request.	
Complete	Package/Request Package \$TMP Transport Request Cocal Object	
	🔁 Back 🛃 Continue 🗶 Cancel	

• On the final step choose *Complete*.

	X
 Object Type Choose Endpoint Choose Operations Configure Service Enter Package/Request △ Complete 	The Web Service ZCustomerCarRental_99 will be created. Transaction WSADMIN (<i>Administration Web Services for SOAP</i> <i>Runtime</i>) supports you with the UDDI registration of the Web Service.
	🛃 Back 🍫 Complete 🗶 Cancel



• You can optionally explore the Service Definition that was generated by the Wizard. Display Service Definition ZCustomerCarRental_99

← → 🎾 😒 📽 🖓 🚬 🖷 🛱 🗮 🔲 🖸	13 🖸			
MIME Repository		Service Definition	ZCustomerCarRental 99 Active	
📇 Repository Browser	ARepository Browser		Internal View Types Used Objects Configuration	WSDL
Tepository Information System				
E Tag Browser		▼☆□₿₿		
异 Transport Organizer		Objects		
GTest Repository		V CustomerCarRental_99	Customer Car Rental Services	
Senterprise Services Browser		🗢 😤 Zcustomercreate99	Create Customer	
		🚽 🗈 🕨 Input		
Pauluun D		Output		
Package		🗢 🧣 Zcustomerfindbyname99	Find Customer by Name	
AP_00_SERVICEENABLEMENT		🗈 🕨 Input		
		Output		
		🗢 😵 Zcustomerreservecar99	Reserve Car for Customer	
Object Name	Description	i D D Input		
TMP ABAP_00_99		Output		
Class Library		Zcustomergetreservations	39 Get List of Customer Reservations	
Function Groups		D D Input		
	Function Group for ABAP 00 - Group	up 🖒 🕩 Output		
Function Modules				
ZCUSTOMERCREATE_99	Create Customer			
ZCUSTOMERFINDBYNAME_99	Find Customer by Name			
ZCUSTOMERGETRESERVATIONS_99	Get List of Customer Reservations			
2COSTOWERRESERVECAR_99	Reserve Carlor Customer			
Service Brouider				
7CustomerCorRental 00	Customer Car Pental Services			

9. To start the SOA Manager, use the transaction code SOAMANAGER. (enter /n SOAManager in the command field of the system task bar). SOA Manager is used to complete the configuration of service providers and consumer proxies in a local ABAP system.

<u>M</u> enu	Edit	<u>F</u> avorites	Extr <u>a</u> s	System	<u>H</u> elp
🖉 🖊 Ins	oamana	ager		۱ 🛛 ۱	C 🙆
SAP E	Easy	/ Acces	s - l	Jser m	ienu
B	2	🔓 Other	menu	*	

• The transaction **SOAMANAGER** should launch a Web Dynpro ABAP based application in your Internet Browser. Choose the *Service Administration* tab and then click on *Single Service Administration*.

Monitoring Tools

• In the Web Service Administration screen that comes up, you can search for your Service Definition (Hint use wildcard **Z*****XX** where XX is your group number)

Web Service Administration				
Search Design Time object for Web Service Configuration				
Search Browse				
Search By Service, Consumer Proxy or Service Group: Search by: Service Search Pattern: Z*99	Field: External Name Tin System: M30/800 Oo Show Advanced Se	arch		
Search Results				
External Name	Namespace	Internal Name		
Q ZCustomerCarRental_99	urn:sap-com:document:sap:soap:functions:mc-style ZCUSTOMERCARRENTAL_			



• Select the row in the Search Results for your Service Definition and then click the *Apply Selection button*. The bottom half of the screen will now show the details for the selected Service Definition.

Search Results		
External Name		Namespace
🗾 🔹 🕘 ZCustomerCarRenta	al_99	urn:sap-com:document:sap:soap:functions:mc-style
Apply Selection		

• There are lots of changes that can be made to the Service Definition from this screen. You can see a summary of all the settings from the *Details* tab. You can alter the published classification settings from the *Classification* tab. From the *Configurations* tab, you see the Endpoints for this service. If you had forgotten to Deploy Service during the wizard, you would now have to create this Endpoint manually. The settings for the Endpoint were generated for us based upon the security profile we choose during the wizard.

<u>o search</u>			
verview Configurations Classificat	ions Details		
sate Endpoint Delete Endpoint Edit Displ	lay	State	Endocinte
3611106		State	Enapointa
ZCUSTOMERCARRENTAL_99		Active	ZCustomerC
Service Configuration of Service Definition	n: ZCUSTOMERCARRENTAL_99		
o Design Time Details			
Save Cancel			
iguration of Web Service 'ZCUSTOMERCAR	RENTAL_99': Endpoint 'ZCustomerCarRental_99'		
ditional Information Security Trans	sport Settings Operation specific		
raneport Security			
Tanaport accurity			
ecurity Mechanism:	Transport Security		
ecurity Mechanism: ransport Protocol:	Transport Security HTTP		
iecurity Mechanism: ransport Protocol: inore SSL Server Certificate:	Transport Security HTTP Ignore SSL Server Certificate		
ecurity Mechanism: ransport Protocol: nore SSL Server Certificate: ignature Expected:	Transport Security HTTP Ignore SSL Server Certificate No Signature Expected		
ecurity Mechanism: ransport Protocol: nore SSL Server Certificate: grature Expected: rustworthy Certificate PSE:	Transport Security HTTP Ignore SSL Server Certificate No Signature Expected WSSKEYS		
iecurity Mechanism: ransport Protocol jnore SSL Server Certificate: ignature Expected: rustworthy Certificate PSE: attern for Certificates:	Transport Security HTTP Ignore SSL Server Certificate No Signature Expected WSSKEYS Subject=",Issuer=",SerialNumber="		
iecurity Mechanism: ransport Protocol ignature Expected ignature Expected rustworthy Certificate PSE: attern for Certificates: igned Message Elements:	Transport Security HTTP Ignore SSL Server Certificate No Signature Expected WISSKEYS Subject=*Issuer=*,SerialNumber=*		
ecurity Mechanism: ransport Protocol ignature Expected: uteworthy Certificate PSE: attern for Certificate PSE: igned Message Elements: norcytion Expected.	Transport Security HTTP Ignore SSL Server Certificate No Signature Expected WSSKEYS Subject=*lssuer=*,SerialNumber=* No Encryption Expected		
iecurity Mechanism: ransport Protocol jorer SSL Server Certificate: ignature Expected: ustworthy Certificate SE: attern for Certificates: igned Message Elements: norryption Expected; dd Signature:	Transport Security HTTP Ignore SSL Server Certificate No Signature Expected WSSKEVS Subject=",issuer=",SerialNumber=" No Encryption Expected No Not Add Signature		
ecurity Mechanism. ransport Protocol nore SSL Server Certificate: ignature Expected rustworthy Certificate PSE: attern for Certificates: igned Message Elements: noryption Expected: dd Signature: SS Signature Key:	Transport Security HTTP Ignore SSL Server Certificate No Signature Expected WSSKEYS Subject=",ssuer=",SerialNumber=" No Encryption Expected No Not Add Signature WSSKEY		
ecurity Mechanism: ransport Protocol janature Expected: janature Expected: attern for Certificate PSE: attern for Certificates: jagned Message Elements: corcytion Expected: dd Signature: SE Signature: Key: jagned Message Elements:	Transport Security HTTP Ignore SSL Server Certificate No Signature Expected WSSKEYS Subject=*,Issuer=*,SerialNamber=* No Encryption Expected No Not Add Signature WSSKEY		
iecurity Mechanism: increptor Protocol jorer SSL Server Certificate: ignature Expected: ustworthy Certificates: attern for Certificates: attern for Certificates: igned Message Elements: ncryption Expected: dd Signature: SE Signature Key: iigned Message Elements: ncryption:	Transport Security HTTP Ignore SSL Server Certificate No Signature Expected WSSKEYS Subject="tesuer=";SerialNumber=" No Encryption Expected No Not Add Signature WSSKEY Do Not Encrypt Message Do Not Encrypt Message		
ecurity Mechanism. ransport Protocol ignature Expected. attern for Certificate: ignature Expected. attern for Certificate SSE: attern for Certificates: igned Message Elements: morphion Expected. dd Signature: SS Signature Key: igned Message Elements: norsystom: SE of Key:	Transport Security HTTP Ignore SSL Server Certificate No Signature Expected WSSKEYS Subject=*Jssuer=*,SerialNumber=* No Encryption Expected No Not Add Signature WSSKEY Do Not Encrypt Message WSSKEY		
ecurity Mechanism: ransport Protocol jonare SSL Server Certificate: janature Expected. attern for Certificate PSE: attern for Certificates: igned Message Elements: neryption: SE of Key: neryption: SE of Key: nioue X 5000 IC/NN:	Transport Security HTTP Ignore SSL Server Certificate No Signature Expected WSSKEVS Subject=*,Issuer=*,SeriaNumber=* No Encryption Expected No Not Add Signature WSSKEV Do Not Encrypt Message WSSCRT		
ecurity Mechanism: ransport Protocol nore SSL Server Certificate: ignature Expected rustworthy Certificate PSE: dittern for Certificates: igned Message Elements: norytoinn Expected: dd Signature: SE Signature Key: igned Message Elements: norytoinn: SE of Key: br:	Transport Security HTTP Ignore SSL Server Certificate No Signature Expected WSSKEYS Subject="tesuer=";SerialNumber=" No Encryption Expected No Not Add Signature WSSKEY Do Not Encrypt Message WSSCRT Signature Certificate		
ecurity Mechanism: ransport Protocol inore SSL Server Certificate: ignature Expected: attern for Certificates: igned Message Elements: noryblion: SE Signature: SE	Transport Security HTTP Ignore SSL Server Certificate No Signature Expected WSSKEYS Subject=*,Issuer=*,SeriaNumber=* No Encryption Expected No Not Add Signature WSSKEY Do Not Encrypt Message WSSCRT Signature Certificate		
ecurity Mechanism: ransport Protocol janature Expected: lignature Expected: attern for Certificate PSE: attern for Certificates: ligned Message Elements: neryption: SE of Key: neryption: SE of Key: New: SE of Key: New: New: SE of Key: New: New: SE of Key: New: New: SE of Key: New: New: New: SE of Key: New: N	Transport Security HTTP Ignore SSL Server Certificate No Signature Expected WSSKEVS Subject=*,Issuer=*,SeriaNumber=* No Encryption Expected No Not Add Signature WSSKEY Do Not Encrypt Message WSSCRT Signature Certificate No Message Age		
ecuity Mechanism: ransport Protocol nore SSL Server Certificate: ignature Expected rustworthy Certificate SSE igned Message Elements: neryption Expected: dd Signature: SS Signature Key: igned Message Elements: neryption: SE of Key: nique X.500 ID (DN): ey: nique X.500 ID (DN): ey: nique A.500 ID (DN): ey: ey: ey: ey: ey: ey: ey: ey	Transport Security HTTP Ignore SSL Server Certificate No Signature Expected VKSSKEVS Subject=*,Issuer=*,SerialNumber=* No Encryption Expected No Not Add Signature VKSSKEV Do Not Encrypt Message VKSSKEV Signature Certificate No Message Age 180		
ecurity Mechanism: ransport Protocol inore SSL Server Certificate: ignature Expected: attern for Certificate PSE: attern for Certificates: igned Message Elements: noryblion: SE of Key: igned Message Elements: noryblion: SE of Key: noryblion: SE of Key: noryblion: SE of Key: noryblion: SE of Key: Norybeid Message Elements: essage Age Expected: aximum Message Age: Oue Fault Security:	Transport Security HTTP Ignore SSL Server Certificate No Signature Expected WSSKEYS Subject=*,Issuer=*,SerialNumber=* No Encryption Expected No Not Add Signature WSSKEY Do Not Encrypt Message WSSCRT Signature Certificate No Message Age 180 No SOAP Faults		
iecurity Mechanism: ransport Protocol: none SSL Server Certificate: ignature Expected: rustworthy Certificate PSE: attern for Certificate PSE: digned Message Elements: norytoin Expected: dd Signature: SS Signature Key: igned Message Elements: norytoin: SE of Key: norytoin: SE of Key: norytot Message Elements: lessage Age Expected: lawinum Message Age: OAP Fault Security:	Transport Security HTTP Ignore SSL Server Certificate No Signature Expected WSSKEYS Subject=*,Issuer=*,SerialNumber=* No Encryption Expected No Not Add Signature WSSKEY Do Not Encrypt Message WSSCRT Signature Certificate No Message Age 160 No SOAP Faults No MS Secure Conversition		

• Go Back to Design Time Details



- Click on the **Overview** tab
- Click on Open Web Service Navigator for the selected binding



•

Details of Service Definition: ZCUSTOMERCARRENTAL
Back to search Overview Configurations Classifications Details
Object Status: Porttype Namespace: Porttype Name: Internal Name: SOAP Applikation: Package Name:
Selected Binding: Open porttype WSDL document Open WSDL document for selected binding or service Open Web Service navigator for selected binding Display selected findings or Service's WSDL UR
Design Time Documentation WSDL URL for Binding: http://iwdfvm3022.wdf.sap.corp:51080/sap/bc/srt/wsdl/sr
Show WSDL Options

• A Separate Browser Window is started. If you are asked to login, enter the credentials Tester / abcd1234

SCP CettVecace Use D Use D<
The url of wsdl is displayed. Click Next
Web Services Navigator
Service Test Test Scenario
Image: Constraint of the service 2 3 4 Image: Constraint of the service Service Operation Input Parameters Result
Previous Next Invocation Parameters Invoke Add to Test Scenario
Service Information
Search Service Interfaces
Secret Tune: @MSDL

• If prompted for a login to download the wsdl – Please enter your userid and pwd for M30 and click OK



• Select the **ZCUSTOMERCREATE_XX** operation (where XX is your group number) and Click **Next**



•

Web Services Navigator
Service Test Test Scenario
Image: Service Operation Input Parameters Result
Previous Next Invocation Parameters Invoke Add to Test Scenario
Service Information
Operations
Operation Name
A
Zcustomercreate99
Zcustomerfindbyname99
Zcustomergetreservations99
Zcustomerreservecar99

• Enter your Group Number for the Customer Name and Click Next

Service Test Test Scenario
Image: market state Image: market state Service Operation Input Parameters
Previous Next Invocation Parameters Invoke Add to Test Scenario
Service Information
Parameters
Customercreate99: Arcore Scoup_XX

• The system displays the input parameters and the result of the test.



• Click on the **Operation** link



Service Test Scenario
Image: service 2 3 4
Previous Next Invocation Parameters Invoke Add to Test Scenario

Service Information

• Select the ZCUSTOMERFINDBYNAME_XX operation (where XX is your group number) and Click Next

-	1 2 3 4 Service Operation Input Parameters Result
4	Previous Next 🕨 Invocation Parameters Invoke Add to
Se	rvice Information
Dpe	erations
B	Operation Name
ፑ	
	Zcustomercreate99
	Zcustomerfindbyname99
	Zcustomergetreservations99
	Zcustomerreservecar99
	Se Dpu

• Enter your Group Number for the Customer Name Click Next

Service Operation Input Parameters Result
Previous Next Invocation Parameters Invoke Add to
Service Information
Parameters
Import Z Export Edit Z
- 😑 Zcustomerfindbyname99: 🖉

• The system displays the input parameters and the result of the test.

Tree View XML Content HTTP Log	Hesuit XML Content HTTP Log
Export	
- 🔁 Zcustomerfindbyname99: 🖉	 ☐ Zoustomerfindbyname99Response: 𝑘 ☐ Customers: 𝑘 ☐ item (1) 𝑘 ☐ item: 𝑘 ☐ ite: 00000012 Name: Group_XX 𝑘 Return: 1 records

• Important Please make note of your Customer ID (we will use this later).



• Click on the **Operation** link

Service Test Scenario
Image: market state Image: market st
Previous Next F Invocation Parameters Invoke Add to Test Scenario

Service Information

• Select the **ZCUSTOMERRESERVECAR_XX** operation (where XX is your group number) and Click **Next**

Image: market state Image: state Service Operation Input Parameters
Previous Next Invocation Parameters Invoke Add t
Service Information
Operations
Operation Name
8
Zcustomercreate99
Zcustomerfindbyname99
Zeustomergetreservations99
C Zcustomerreservecar99

• Enter a Category A and the CustomerID returned from the ZCUSTOMERFINDBYNAME_XX operation. Please be sure to enter the Start and End dates using the YYYY-MM-DD ISO date format, Click Next

Image: Constraint of the second se
Previous Next I Invocation Parameters I Invoke Add to T
Service Information
Parameters
 Customerreservecar99: Category: * Customerld: * 0000012 Customerld: * 2009-10-02 Enddate: * 2009-10-01 Startdate: * 2009-10-01

• The system displays the input parameters and the result of the test.





Optional – Set an External Breakpoint in Function Module ZCUSTOMERRESERVECAR_XX (where XX is your group number) by right clicking in the left margin and **Set External Breakpoint** after which you can debug from the WS Navigator into the ABAP system

Attributes Import Export Changing Tables Exceptions Source code 16 1_reservation TYPE demo_cr_sreservation. 17 1 18 1r service = cl demo_cr_ar rental service=>get service
16 1_reservation TYPE demo_cr_sreservation. 17 18 18 1r service = cl demo cr car rental service=>get service
16 l_reservation TYPE demo_cr_sreservation. 17 18 18 lr service = cl demo cr car rental service=>get service
17 18 In service = cl demo cr car rental service=>get service
19
Bookmarks make_reservation(
Goto Bookmark + G
More , omer_id = customer_id
<u>Set Session Breakpoint</u> <u>from</u> = startdate
Set External Breakpointto = enddate).
Delete Session Breakpoint cr_no_customer cx_demo_cr_lock cx_dem Delete External Breakpoint exception->get_text(). .
30 CATCH cx root INTO lo exception.
31 return = lo_exception->get_text().
32 EXIT.
33 - ENDTRY.
34
35 l_customer = lr_service->get_customer_by_id(customer_i

• You can step thru the code using and learning how to use it is essential for Java programmers accessing ABAP Business Logic.





© 2009 by SAP AG.

All rights reserved. SAP, R/3, SAP NetWeaver, Duet, PartnerEdge, ByDesign, SAP Business ByDesign, and other SAP products and services mentioned herein as well as their respective logos are trademarks or registered trademarks of SAP AG in Germany and other countries.

Business Objects and the Business Objects logo, BusinessObjects, Crystal Reports, Crystal Decisions, Web Intelligence, Xcelsius, and other Business Objects products and services mentioned herein as well as their respective logos are trademarks or registered trademarks of Business Objects S.A. in the United States and in other countries. Business Objects is an SAP company.

All other product and service names mentioned are the trademarks of their respective companies. Data contained in this document serves informational purposes only. National product specifications may vary.

These materials are subject to change without notice. These materials are provided by SAP AG and its affiliated companies ("SAP Group") for informational purposes only, without representation or warranty of any kind, and SAP Group shall not be liable for errors or omissions with respect to the materials. The only warranties for SAP Group products and services are those that are set forth in the express warranty statements accompanying such products and services, if any. Nothing herein should be construed as constituting an additional warranty.

