

# Lecture 6 Workbook

## ABAP objects, Design patterns

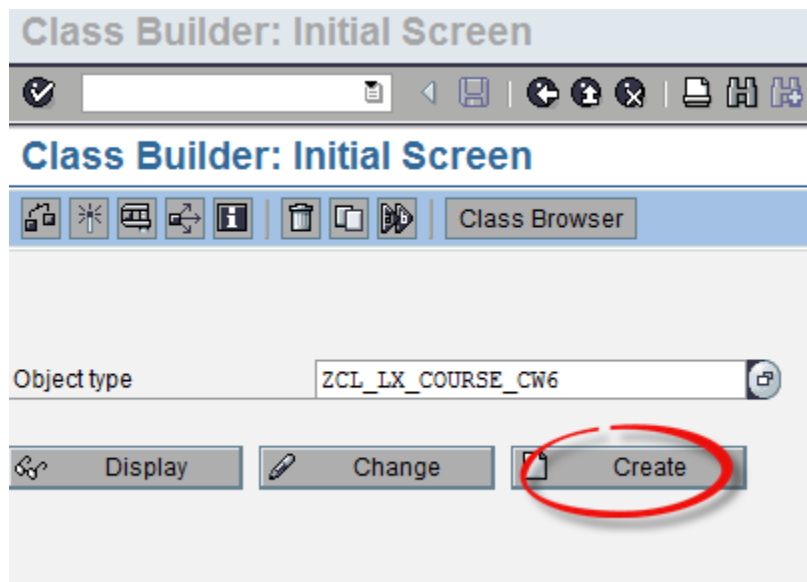
### Contents

HOW TO CREATE A CLASS.....	2
HOW TO REALIZE A SINGLETON PATTERN .....	5

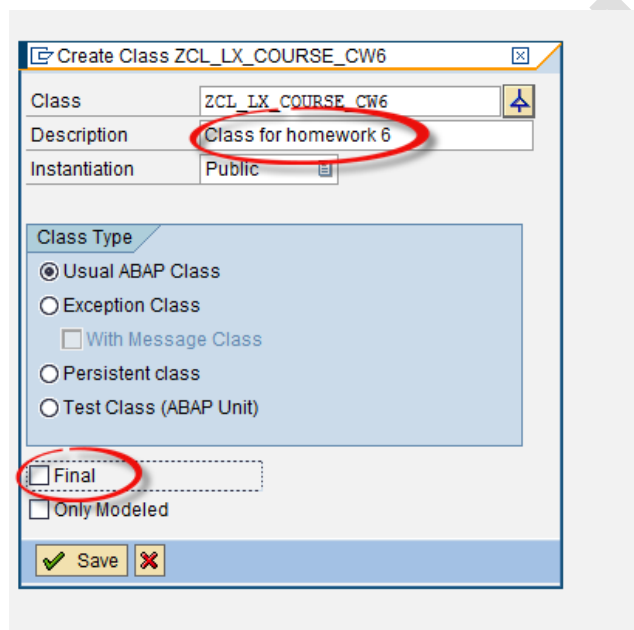
## How to create a class

Classes are templates for objects. Conversely, you can say that the type of an object is the same as its class. A class is an abstract description of an object. You could say that it is a set of instructions for building an object. The attributes of objects are defined by the **components** of the class, which describe the state and behavior of objects.

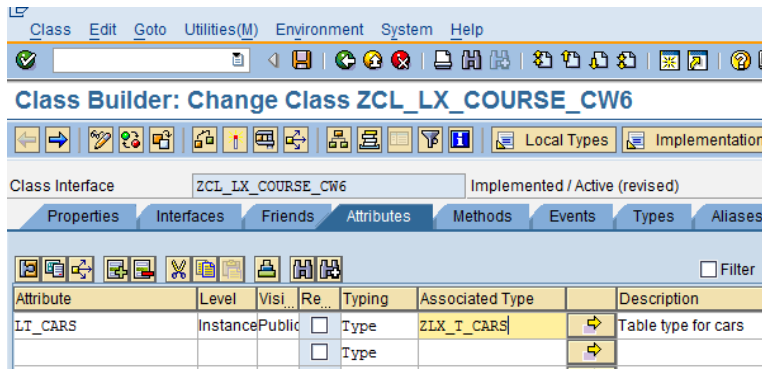
Go to transaction SE24. Provide Object type like ZCL\_XXX and click on Create.



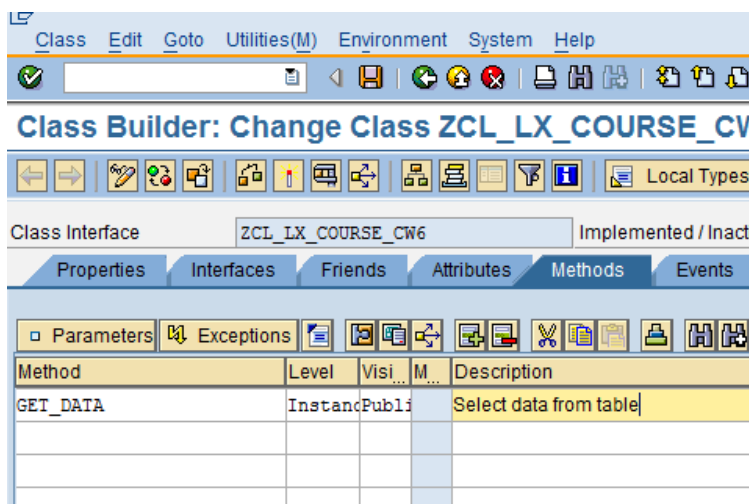
Provide Description and unmark Final. Save it.



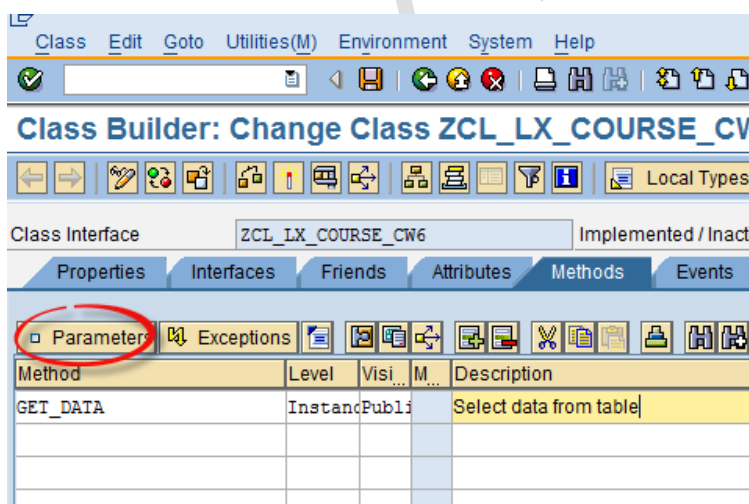
Now you can see Class Navigator.  
Click on the tab Attributes. Provide necessary attributes.

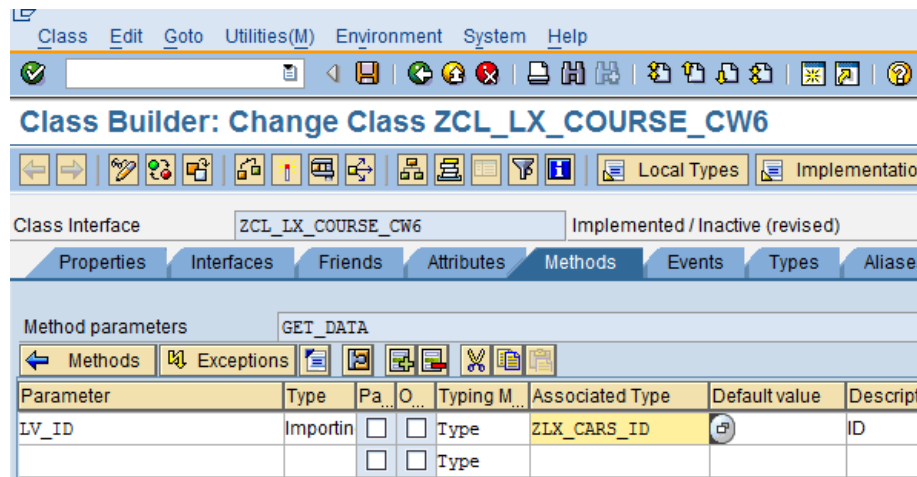


On the Methods tab provide name of the method for example GET\_DATA.



Select method and click on Parameters. Provide name of parameters. Save and Activate it.





Double click on method's name. Enter source code. Save and Activate it.  
Go to main program. Now you need create an instance and call your methods.

#### Create instance.

```
DATA go_cars TYPE REF TO zcl_lx_course_cw6.
CREATE OBJECT go_cars.
```

#### Call method.

```
CALL METHOD go_cars->get_data
EXPORTING
    lv_id    = <value>.
```

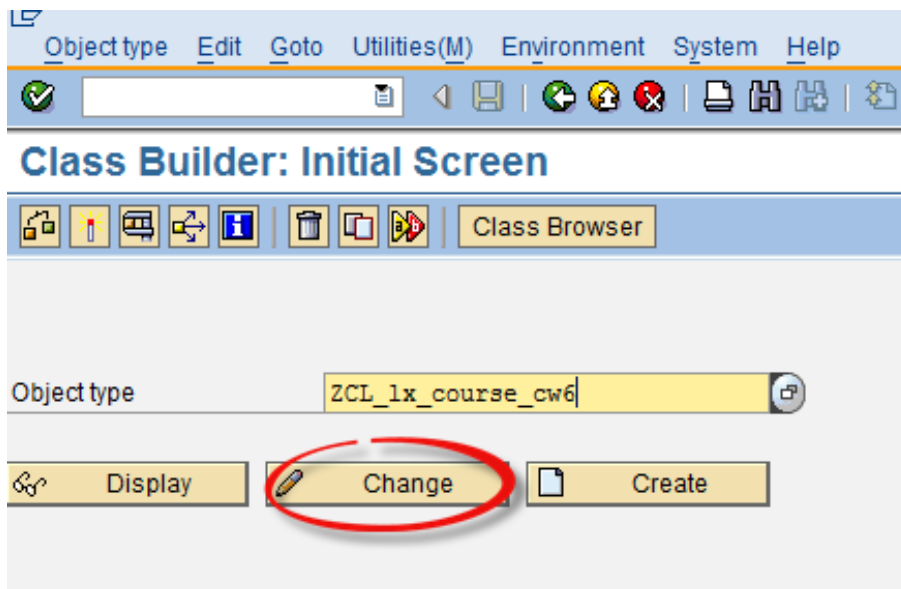
## How to realize a singleton pattern

In software engineering, the singleton pattern is a design pattern that restricts the instantiation of a class to one object. This is useful when exactly one object is needed to coordinate actions across the system. The concept is sometimes generalized to systems that operate more efficiently when only one object exists, or that restrict the instantiation to a certain number of objects. The term comes from the mathematical concept of a singleton.

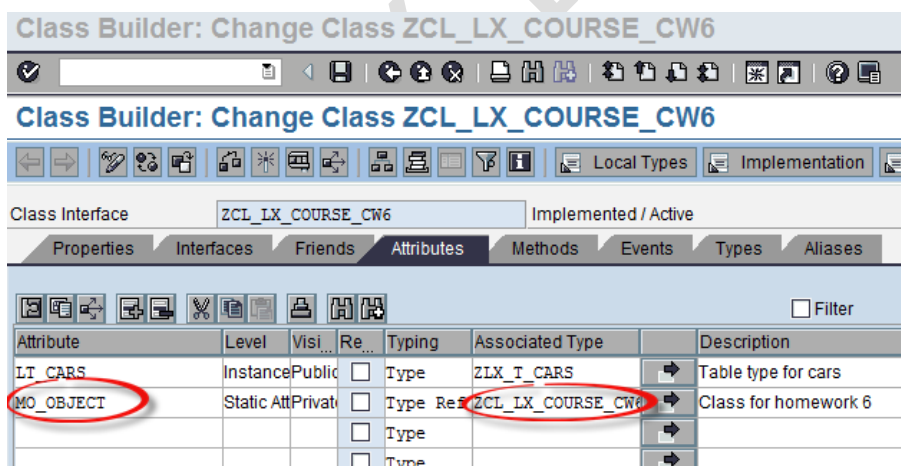
A class which implements Singleton Pattern should have the following things other than the class's own data members and methods:

1. Private constructor(s)
2. A private static member of the same type of the class.
3. A static method which returns the only instance.

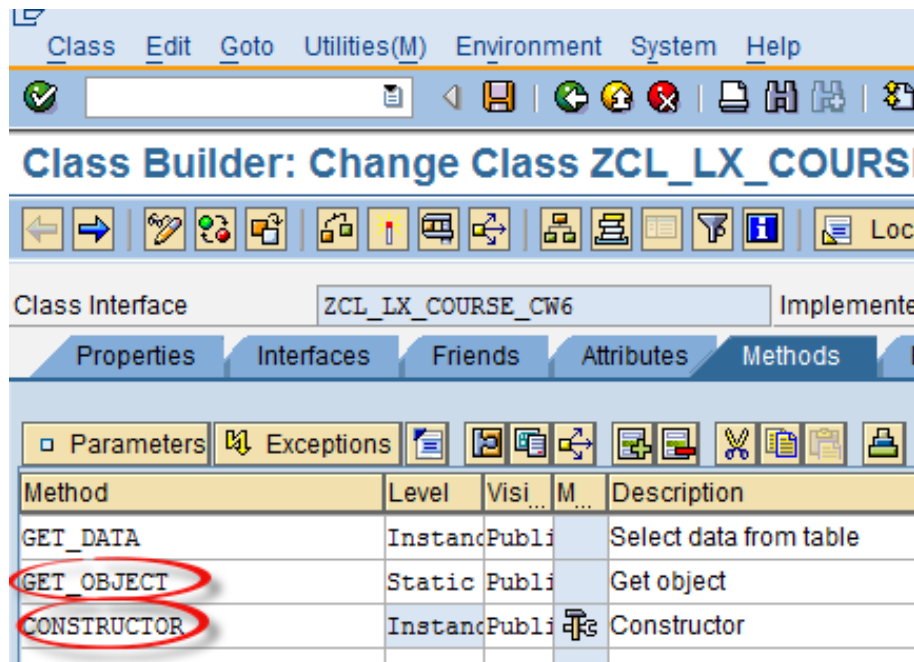
Go to transaction SE24, select your class, press Change.



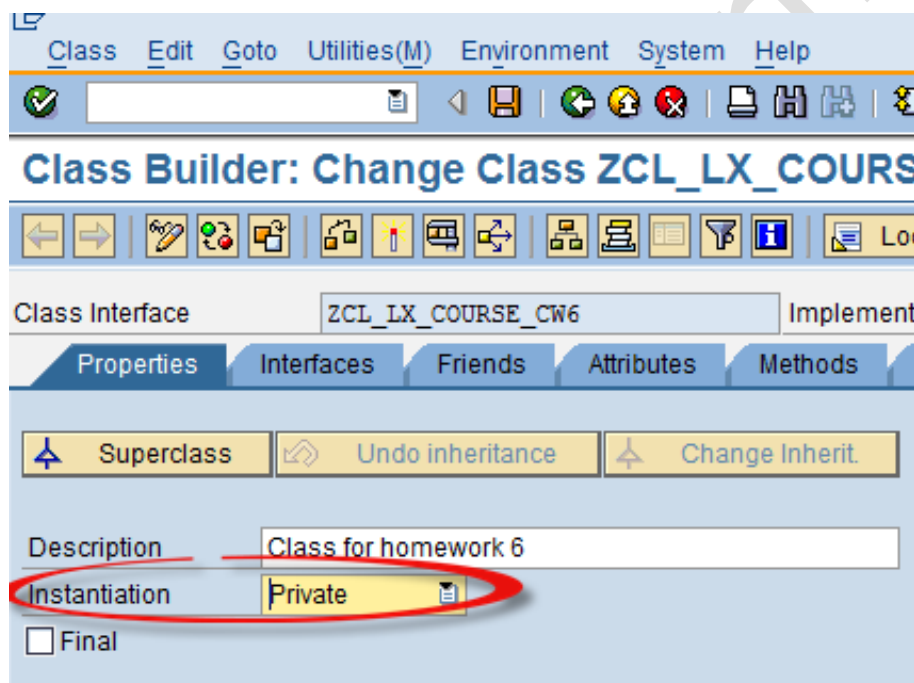
On tab Attributes add static attribute of the class.



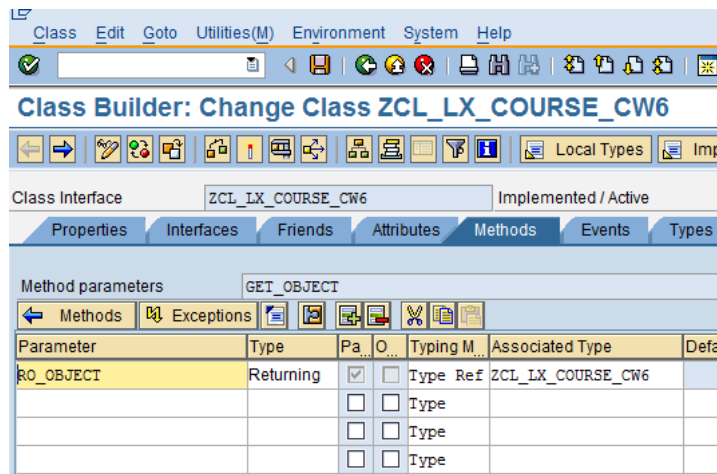
On the tab Methods provide one static method and one constructor.



We need to create private constructor. Open tab Properties and choose Instantiation -> Private



Create an attribute for method GET\_OBJECT. This attribute should have Type: Returning, Associated Type: <your class>.



Now provide source code for method GET\_OBJECT.

```
IF mo_object IS INITIAL.
  CREATE OBJECT mo_object.
ENDIF.
ro_object = mo_object.
```

Provide source code for CONSTRUCTOR. For example:

```
SELECT zlxcars~id zlxcars~vin zlxcars~licplate zlxcars~naxles
       zlxcars~customer knal~name1 zlxcarst~descr
INTO CORRESPONDING FIELDS OF TABLE lt_cars
FROM zlxcars INNER JOIN zlxcarst
     ON zlxcars~id = zlxcarst~id INNER JOIN knal
     ON zlxcars~customer = knal~kunnr.
```

Save and Activate it.


Now if you want to create instance of your class you need call static method:

```
DATA lo_cars TYPE REF TO ZCL_LX_COURSE_CW6.
lo_cars = ZCL_LX_COURSE_CW6=>GET_OBJECT( ).
```

If you try to create instance of the class with constructor

```
create OBJECT lo_cars.
```

You will see error message

Description	Row	Type
Program ZLX_CW6	40	
You cannot create an instance of the class "ZCL_LX_COURSE_CW6" outside the class..		