

Applies To:

ABAP 4.6C

Summary

The code sample depicts the Object Oriented concept of ABAP language.

By: Subathra Radhakrishnan

Company and Title: Wipro Technologies

Date: 22 Feb 2005

Table of Contents

CODE:

report zoops2 .

* CLASS SHAPE DEFINITION

class shape definition.

public section.

 methods: get_area.

protected section.

 data area type p decimals 4.

 methods: get_data.

endclass.

* CLASS SHAPE IMPLEMENTATION

class shape implementation.

method get_data.

write:/ ' Area of circle(from protected method):',area.

endmethod.

method get_area.

write: area.

*A method in the protected section cannot be directly called using an

*object.It can be called from within a public method as below.

call method get_data.

endmethod.

endclass.

* CLASS CIRCLE DEFINITION

* In ABAP classes inherit using "inheriting from" keyword.

class circle definition inheriting from shape.

public section.

methods: set_radius importing value(set_value) type i,

get_area redefinition.

private section.

data radius type i.

data pi type p decimals 4.

endclass.

```
*-----*  
*   CLASS CIRCLE IMPLEMENTATION  
*-----*
```

class circle implementation.

method set_radius.

pi = '3.14'.

radius = set_value.

endmethod.

method get_area.

area = pi * radius * radius .

write : 'AREA OF CIRCLE(showing polymorphism): '.

* Calling Super Class Method (Shape class) to show Polymorphism

call method super->get_area.

endmethod.

endclass.

start-of-selection.

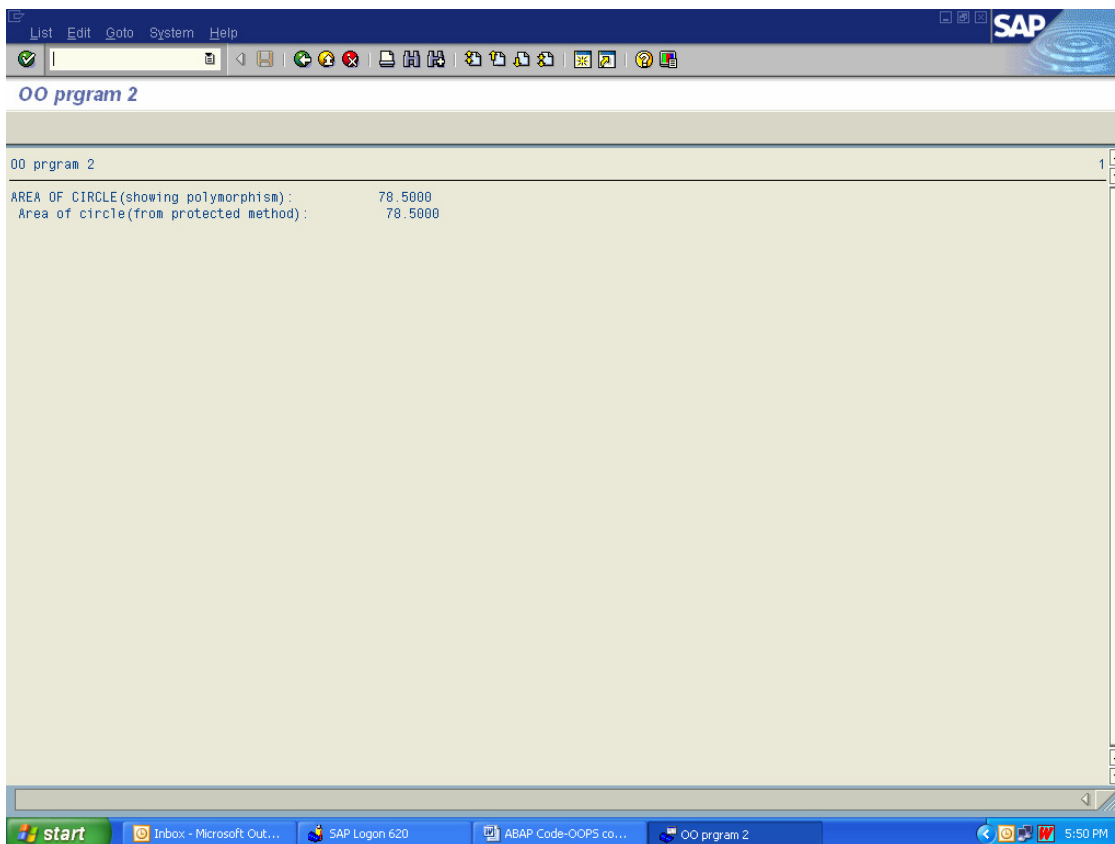
data cref type ref to circle.

create object cref.

call method cref->set_radius exporting set_value = 5.

call method cref->get_area.

OUTPUT SCREEN:



Disclaimer & Liability Notice

This document may discuss sample coding, which does not include official interfaces and therefore is not supported. Changes made based on this information are not supported and can be overwritten during an upgrade.

SAP will not be held liable for any damages caused by using or misusing of the code and methods suggested here, and anyone using these methods, is doing it under his/her own responsibility.

SAP offers no guarantees and assumes no responsibility or liability of any type with respect to the content of the technical article, including any liability resulting from incompatibility between the content of the technical article and the materials and services offered by SAP. You agree that you will not hold SAP responsible or liable with respect to the content of the Technical Article or seek to do so.

Author Bio

I am working as an ABAP programmer in Wipro Technologies.

Copyright © 2005 SAP AG, Inc. All Rights Reserved. SAP, mySAP, mySAP.com, xApps, xApp, 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 in several other countries all over the world. All other product, service names, trademarks and registered trademarks mentioned are the trademarks of their respective owners.