

## ABAP接口与事件

interface output. 接口的定义

data hello(30) type c .

methods write.

endinterface.

class superclass definition.

public section.

interfaces output.接口在类中实现 ( 只能在public )

events evt exporting value(p1) type i.

methods m1.

events return.申明事件，可以在类和接口中

methods accumulate.

class-data num type i.申明静态变量

data a1 type i value 7.

endclass.

class superclass implementation.类方法实现

method output~write.在类中实现接口方法

output~hello = 'hello, sap is coming!'.

write:/ 'hello,sap!'.

write:/ output~hello.

endmethod.

method m1.

raise event evt exporting p1 = a1.

endmethod.

method accumulate.

data: res type i.

num = num + 1.

res = num mod 10.

if res = 0.

raise event return.触发事件return

endif.

endmethod.

endclass.

class handler definition.

public section.

methods handler\_execute for event return of superclass.

endclass.

class handler implementation.

method handler\_execute.

write:/ 'the persent number is', superclass=>num.

endmethod.

endclass.

data handler\_obj type ref to handler.

data: class\_obj type ref to superclass,引用类

intf\_obj type ref to output,引用接口

intf\_table type table of ref to output.引用接口创建内表

start-of-selection.

create object: class\_obj, handler\_obj.创建类的实例

call method class\_obj->output~write.通过类的实例调用接口的方法

loop at intf\_table into intf\_obj.通过接口引用调用接口方法

write:/ sy-index.

call method intf\_obj->write.

endloop.

set handler handler\_obj->handler\_execute for all instances.注册事件

do 50 times.

call method class\_obj->accumulate.循环调用，满足条件时触发事件return

enddo.

---

接口与抽象类的区别：

1，抽象类拥有抽象成员，也可以拥有非抽象成员，接口全部是抽象成员

2，抽象类成员可以是私有的，接口一般都是公有的

3，接口中不能含有构造方法，静态成员和常量

接口没有自己的实例，方法的实现要通过具体的类进行，一个接口可以被任意个类实现。在类的定义中，接口只能申明为公有，接口中所有的组件被添加为该类的公有组件。接口方法的调用可以通过类的实例来实现，格式obj->interface->method,如果要通过接口引用直接调用该方法，必须先创建接口引用的内表，通过oop对象执行

