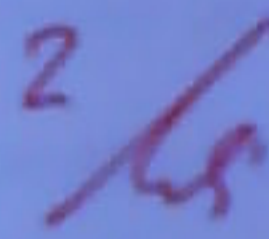


Klausur
BC401

Name: SPATHIA
Vorname: Fatma

Note: 89 $\frac{1}{2}$ 2
12/16 

1. Welche Aussagen treffen zu:

- Bei der Definition von Klassen kann man für die Sichtbarkeit von Attributen drei Bereiche anlegen:
- PUBLIC SECTION
 - PROTECTED SECTION
 - PRIVATE SECTION.
- Unterklassen können nur auf Attribute zugreifen, die in der übergeordneten Klasse in der PUBLIC SECTION stehen.
- Bei der Vererbung von Klassen können in den Unterklassen zusätzliche Methoden definiert und implementiert werden.
- Es ist nicht möglich, Methoden der übergeordneten Klasse in der Unterklasse mit einer neuen Funktionalität zu versehen.
- Beim Erzeugen eines neuen Objektes (CREATE OBJECT) zu einer Unterklasse deren übergeordnete Klasse einen Instanz-Constructor besitzt, müssen auch die Importparameter des Instanz-Constructors der übergeordneten Klasse als Exportparameter angegeben werden .
- Unterklassen dürfen keinen eigenen Instanz-Constructor besitzen.
- Der Instanz-Constructor ist eine Methode, die automatisch beim Anlegen eines Objektes aufgerufen wird.
- Statische Methoden (Klassen-Methoden) werden im Programm wie folgt aufgerufen:
- CALL METHOD <REFERENZVARIABLE>-><METHODENNAME>
z.B:
DATA zeiger TYPE REF TO cl_kunden.
CREATE OBJECT zeiger EXPORTING im_name = 'Meier'.
CALL METHOD zeiger->anzeige.
- UML (unified modeling language) ist eine von der SAP entwickelte standardisierte Modellierungssprache.
- Eine funktionale Methode ist eine Methode, die einen Returning-Parameter besitzt.

2. Welche Aussagen treffen zu:

- Für funktionale Methoden können zusätzlich zum Returning-Parameter Exporting- und Importing-Parameter angegeben werden.
- Der folgende Programmausschnitt enthält keine Fehler:

```
CLASS cl_kunde DEFINITION.  
  PUBLIC SECTION.  
  METHODS: set_attributes IMPORTING im_name TYPE knal-name1  
           im_ort TYPE knal-ort01.  
           CLASS-DATA: count TYPE I.  
  PROTECTED SECTION.  
  METHODS: constructor IMPORTING im_name TYPE knal-name1  
           im_ort TYPE knal-ort01.  
  PRIVATE SECTION.  
  DATA: name TYPE knal-name1,  
        ort TYPE knal-ort01.  
ENDCLASS.
```

CONSTRUCTOR MUSS IMMER PUBLIC SEIN!!!!

- 2

Sollte ein Fehler in diesem Programmausschnitt sein, kringeln Sie bitte die Fehlerstelle ein.

- CL_SUPER soll eine Klasse sein, von der die Unterklasse CL_SUB abgeleitet wurde. Durch

```
DATA: R_SUPER TYPE REF TO CL_SUPER,  
      R_SUB TYPE REF TO CL_SUB.
```

werden je ein Zeiger auf die beiden Klassen definiert. Nach den Anweisungen

```
CREATE OBJECT r_super.  
CREATE OBJECT r_sub.
```

kann der Zeiger R_SUB durch die Anweisung

```
R_SUB = R_SUPER.
```

R_SUB ?= R_SUPER

so gecastet (umgewandelt) werden, dass er auf das Objekt der Klasse CL_SUPER zeigt.

- Finale Klassen können keine Unterklassen besitzen.
- Um in Unterklassen Methoden die von der übergeordneten Klasse geerbt wurden eine neue Funktionalität zu geben, müssen diese Methoden in der Unterklasse redefiniert werden.

3. Kreuzen Sie die richtigen Antworten an

- Beim Redefinieren einer Methode muss zu der Klasse, in der die Methode neu geschrieben wird, ein zusätzlich zum Definitions- und Implementationsteil ein Redefinitionsteil

CLASS <klassenname> REDEFINITION.

...
ENDCLASS.

angelegt werden

- Wenn Objekte unterschiedlicher Klassen auf gleiche Methodenauftrufe unterschiedlich reagieren, spricht man von Polymorphie.
- Polymorphie sollte beim Programmieren möglichst vermieden werden, weil dadurch die Nutzung von Klassen unnötig erschwert wird.

4. Gegeben ist der folgende Programmausschnitt:

```
CLASS flugzeug DEFINITION.  
  PUBLIC SECTION.  
    METHODS: constructor IMPORTING im_flugzeug TYPE C.  
  PRIVATE SECTION.  
    DATA: flugzeug TYPE C.  
ENDCLASS.
```

```
CLASS flugzeug IMPLEMENTATION.  
  METHOD constructor.  
    flugzeug = im_flugzeug.  
  ENDMETHOD.  
ENDCLASS.
```

```
DATA: r_flug TYPE REF TO flugzeug.  
      it_tab TYPE TABLE OF REF TO flugzeug.  
CREATE OBJECT r_flug EXPORTING im_flugzeug = 'BOING'.  
append r_flug to it_tab.  
CREATE OBJECT r_flug EXPORTING im_flugzeug = 'AN24'.  
append r_flug to it_tab.
```

Der Inhalt der internen Tabelle it_tab ist:

-

BOING
AN24

Zeiger auf Objekt 'BOING'
Zeiger auf Objekt 'AN24'

-A

14.09.07 10:45

5. Welche der folgenden Aussagen sind korrekt?

- Statische Komponenten existieren nur einmal pro Klasse.
- Statische Komponenten können nur in der PRIVATE SECTION definiert werden.
- Statische Attribute werden mit der Anweisung CLASS-DATA angelegt.
- Attribute die bei der Instanziierung an die Objekte übergeben werden, heißen dynamische Attribute. Sie werden mit der Data-Anweisung deklariert.
- Auf statische Attribute kann, unabhängig davon, ob bereits ein Objekt ihrer Klasse angelegt wurde, mit folgender Syntax zugegriffen:
<Objektreferenz>-><Name des statischen Attributes>
- Auf statische Attribute einer Klasse kann auch zugegriffen werden, wenn noch kein Objekt zu dieser Klasse angelegt wurde
- Statische Methoden können nur auf statische Komponenten zugreifen. Auf Instanzkomponenten haben statische Methoden keinen Zugriff.

6. Kann eine Klasse auch ohne Konstruktor definiert werden?

- ja
- nein

7. Angenommen, Sie kopieren eine Unterklassenreferenz auf eine Referenzvariable, die auf die Oberklasse typisiert ist (Narrowing Cast). Auf welche Komponenten können Sie über diese Referenzvariable zugreifen?

Wählen Sie die richtige(n) Antwort(en).

- geerbte Komponenten der Oberklasse
- redefinierte Komponenten der Oberklasse
- neu definierte Komponenten der Unterklasse

8. Hat ein Interface einen Implementationsteil?

- ja
- nein

- 3

9. Welche der folgenden Aussagen sind korrekt?

Wählen Sie die richtige(n) Antwort(en).

- Eine programmlokale Klasse kann im *Class Builder* in eine globale Klasse umgewandelt werden.
- Eine programmlokale Klasse kann im *Class Builder* kopiert werden. Bei der Kopie handelt es sich dann um eine globale Klasse.
- Mit dem *Class Builder* kann man Funktionsbausteine anlegen.
- Mit dem *Refactoring Assistant* können Sie innerhalb einer Vererbungshierarchie Methoden in andere Klassen verschieben.

10. Um den Benutzer eine objektorientierte Anwendung ausführen lassen zu können, muss immer ein ausführbares, ein Modulpool- oder ein Funktionsgruppenprogramm existieren, da andernfalls nirgendwo eine Instanz per CREATE OBJECT erzeugt werden könnte.

Geben Sie an, ob diese Aussage richtig oder falsch ist.

- Richtig
- Falsch

11. Welche der folgenden Aussagen sind korrekt?

Wählen Sie die richtige(n) Antwort(en).

- Der .Freund. einer Klasse ist auch Freund deren Unterklassen.
- Die Unterklassen des Freundes einer Klasse sind ebenfalls Freunde der Klasse.
- Die Sichtbarkeit eines Instanzkonstruktors kann eingeschränkt werden.
- Die Definition eines privaten Instanzkonstruktors (Instanzierbarkeit nur durch die Klasse selbst) kann innerhalb des privaten Sichtbarkeitsbereiches vorgenommen werden (PRIVATE SECTION).
- Eine nicht-abstrakte Klasse kann abstrakte Methoden enthalten.
- Eine abstrakte Klasse enthält keine Implementationen.
- Eine abstrakte Methode enthält keine Implementation.
- Abstrakte Methoden können nicht redefiniert werden.

- Finale Klassen besitzen keine Unterklassen.
- Eine finale Methode muss redefiniert werden.
- Die Methoden einer finalen Klasse sind immer finale Methoden.

17. Legen Sie eine Klasse CL_FLUGZEUG an.
 Die Klasse soll einen Instanz-Konstruktor besitzen, der folgende Importparameter übernimmt:
 im_flugzeugname
 im_flugzeugtyp.

Weiterhin soll eine Methode ANZEIGE definiert und implementiert werden die den FLUGZEUGNAMEN und den FLUGZEUGTYP ausgibt.

Class Leb-flugzeug Definition.
 Public Section.

Methods:

constructor
 IMPORT: 26
 im_flugzeugname TYPE string
 im_flugzeugtyp TYPE saplane-planetype.
 display-attribute.

Class-Methods:

class constructor.

?

Private Section.

DATA: flugzeugname TYPE string.
 flugzeugtyp TYPE saplane-planetype.

Class-DATA: List-of-plane type TYPE TABLE OF
 Saplane WITH KEY plane type.
 ENDCCLASS.

?

18. Legen Sie eine Unterklasse CL_PASSAGIERFLUGZEUG zur Klasse CL_FLUGZEUG (Aufgabe 17) an.
 Dieser Klasse soll zusätzlich die Anzahl der Sitzplätze übergeben werden.
 Die Methode ANZEIGE soll so geändert werden, dass auch die Anzahl der Sitzplätze ausgegeben wird.

```

class (cl-passagierflugzeug DEFINITION INHERITING
INHERITING FROM (cl-airplane.
Public Section.
METHODS:
  constructor
  importing
    im-flugzeugname-p TYPE string
    im-flugzeugtyp-p TYPE suplane-plane-type
    im-max-seats TYPE s-seatsmax,
  display-attributes REDEFINITION.
PRIVATE SECTION
  DATA: maxseats TYPE s-seatsmax s-seatsmax.
END CLASS.
  
```

siehe Rückseite!!!

19. Erzeugen Sie ein Objekt der Klasse CL_PASSAGIERFLUGZEUG.
 Rufen Sie die Methode ANZEIGE dieser Klasse auf.

```

DATA: r-flugzeug-p TYPE REF TO cl-flugzeug.
      r-passagierflugzeug-p TYPE REF TO cl-passagierflugzeug.
      flugzeug-list TYPE TABLE OF REF TO cl-flugzeug.

START-OF-SELECTION.
CREATE OBJECT r-passagierflugzeug EXPORTING
  im-flugzeugname-p = 'LH BERLIN'
  im-flugzeugtyp-p = '747-400'
  im-seats = 345.
r-passagierflugzeug->display-attributes().
  
```


Class lcl-flugzeug IMPLEMENTATION.

Method constructor.

name = im-name

planetyp = im-planetyp.

ENDMETHOD.

METHOD class_constructor.

SELECT * FROM suplane INTO TABLE lcl-of-planetyp.

ENDMETHOD.

METHOD display-attributes.

WRITE: / 'Name des Flugzeugs: ', name,
 'Flugzeugtyp: ', planetyp.

ENDMETHOD.

ENDCLASS.

Die gekennzeichneten Teile des Programms
sind an der Aufgabenstellung, nicht relevant und
werden in Ihrem Prop. auch nicht benötigt.

-2