



Rewarding Learning

**ADVANCED**

**General Certificate of Education**

**2019**

Centre Number

--	--	--	--	--

Candidate Number

--	--	--	--

# Software Systems Development

Unit AS 1

Introduction to Object Oriented Development



SDV11

**[SDV11]**

**WEDNESDAY 22 MAY, AFTERNOON**

## TIME

2 hours.

## INSTRUCTIONS TO CANDIDATES

Write your Centre Number and Candidate Number in the spaces provided at the top of this page.

Write your answers in the spaces provided in this question paper.

Answer **all seven** questions.

## INFORMATION FOR CANDIDATES

The total mark for this paper is 100.

Figures in brackets printed down the right-hand side of pages indicate the marks awarded to each question or part question.

## ADVICE TO CANDIDATES

You are advised to take account of the marks for each part question in allocating the available examination time.

For Examiner's use only			
Question	Marks available	Marks	Remark
1	9		
2	18		
3	18		
4	20		
5	14		
6	5		
7	16		
<b>Total</b>	<b>100</b>		

- 1 Underline the most appropriate word from those given in the brackets that will correctly complete each statement.

An object is an (**example, instance**) of a class that can perform a set of related activities that define its (**behaviour, operation**).

Abstract classes (**can, cannot**) be instantiated.

An abstract class can only be used as a (**super / base class, interface, derived / subclass**) for other classes that (**change, extend**) it.

A class can inherit from (**one, many**) abstract class(es) and must (**overload, override**) all its methods/properties that are declared to be (**abstract, virtual**) methods/properties.

Polymorphism at run-time is a primary concept of object-oriented programming which is enabled through (**early binding, late binding**).

[9]

Examiner Only	
Marks	Remark

- 2 (a) Methods are fundamental building blocks of programming languages. Give three advantages for their use.

Advantage 1 \_\_\_\_\_

\_\_\_\_\_

Advantage 2 \_\_\_\_\_

\_\_\_\_\_

Advantage 3 \_\_\_\_\_

\_\_\_\_\_ [3]

- (b) Write code to create and populate an array, Months, with the months of the year in abbreviated form.

**Months**

Jan	Feb	Mar	....	....	Nov	Dec
-----	-----	-----	------	------	-----	-----

\_\_\_\_\_

\_\_\_\_\_

\_\_\_\_\_

\_\_\_\_\_ [3]

Examiner Only

Marks

Remark





3 Exception Handling is implemented to ensure that a system will not crash due to user entry error.

(a) Name the basic catch-all exception.

\_\_\_\_\_ [1]

(b) **Describe** how Exception Handling would be implemented to prevent a crash if a date is entered incorrectly.

\_\_\_\_\_  
 \_\_\_\_\_  
 \_\_\_\_\_  
 \_\_\_\_\_  
 \_\_\_\_\_  
 \_\_\_\_\_  
 \_\_\_\_\_  
 \_\_\_\_\_ [5]

(c) Many organisations ensure that their customers are 18 years of age or over to secure a higher level of legal security for the organisation.

### A range of date methods for C# and Java

#### C# DateTime methods:

DateTime.Now.Date	date part of current date;
myDate.AddYears(n)	add or subtract years from a date;

#### Java methods Date format:

Date()	current date;
myDate.before(newDate)	returns true / false
myDate.after(newDate)	returns true / false

#### methods Calendar format:

Calendar.getInstance();	current date
cDate.add(Calendar.YEAR, n);	add or subtract years
// convert Date to Calendar format	
cDate.setTime(currentDate);	where cDate is a Calendar instance
// convert Calendar to Date format	
Date dd = cDate.getTime()	

#### Comparing dates- C#, Java

myDate.CompareTo(newDate)	returns 0 if equal -ve if invoking date less +ve if greater
---------------------------	---

Examiner Only	
Marks	Remark







4 (a) Using examples for illustration describe what you understand by the following class terms.

**overloading**

---

---

---

---

**example**

---

---

---

---

**interface**

---

---

---

---

**example**

---

---

---

---

[8]

Examiner Only	
Marks	Remark

- (b) A class **Book** has been designed to hold data about available books for a school sale.

```
public class Book
{
    private String    bookNo;    //ISBN No
    private String    author;
    private String    title;
    private double    price;
    private int       noCopies;
    private int       noSold;
}
```

- (i) Write code to instantiate an array, arrBook, capable of holding data on 50 books.

---

---

---

---

---

---

---

---

[4]

Examiner Only	
Marks	Remark



- 5 The following class method has been designed to validate a book code made up of ten characters.

```
public boolean validBookCode( string bookCode)
{
    if( validBookFormat( string bookCode)
        if( validBookRelationship( string bookCode)
            return true;
    return false;
}
```

- (a) Giving your reason, select an appropriate visibility option, from the list given below in brackets, for the methods validBookFormat and validBookRelationship. Underline your selection.

( public   private   protected)

Reason \_\_\_\_\_  
 \_\_\_\_\_ [2]

- (b) Write the code for the method validBookFormat() that will return true if the **format** of a book code consists of 10 digits or 9 digits followed by the character X.

\_\_\_\_\_  
 \_\_\_\_\_  
 \_\_\_\_\_  
 \_\_\_\_\_  
 \_\_\_\_\_  
 \_\_\_\_\_  
 \_\_\_\_\_  
 \_\_\_\_\_  
 \_\_\_\_\_  
 \_\_\_\_\_ [4]

Examiner Only	
Marks	Remark

**BLANK PAGE**  
**(Questions continue overleaf)**

(c) Write the code for the method validBookRelationship() that will return true if a remainder of zero is given after the following process:

- Multiply each digit of the book code by the numbers 10 to 1 descending from left to right.  
(If the final character is an X it should be replaced with the number 10 for the process)
- Sum the result of each product.
- Divide the sum by 11.
- Determine if the remainder is zero.

Example 0201530813 is a valid code.

(if character X  
replace with 10)

Book Code	0	2	0	1	5	3	0	8	1	3
Multiply book code digit by	10	9	8	7	6	5	4	3	2	1
Sum products	0+	18+	0+	7+	30+	15+	0+	24+	2+	3 = 99
Divide by 11	$99/ 11 = 9$ <b>Remainder 0</b>									

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

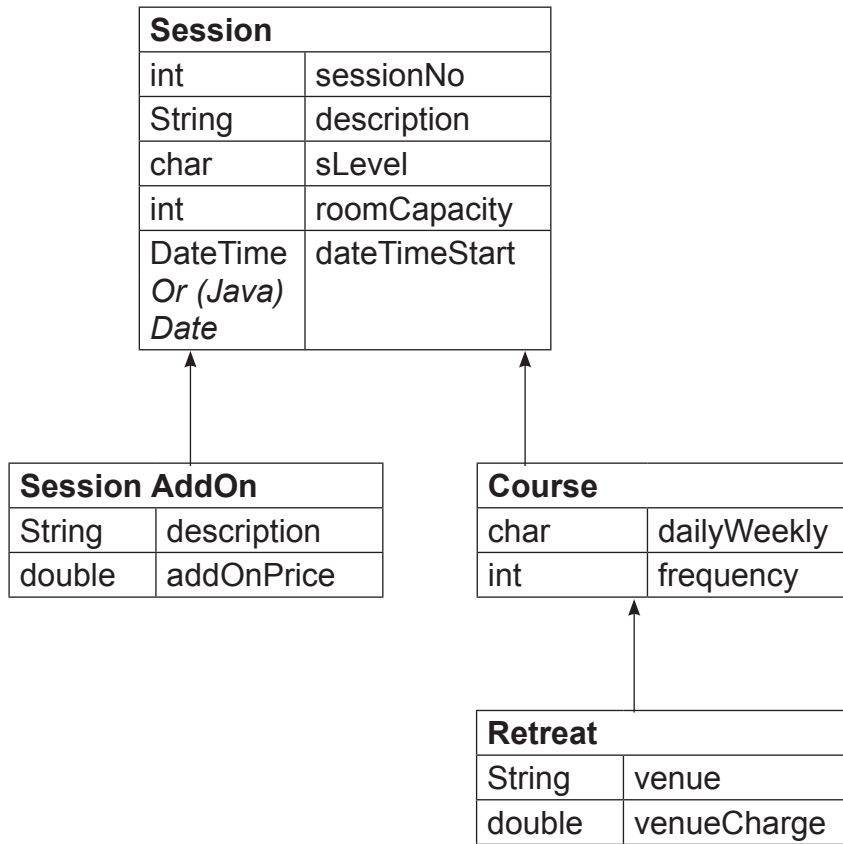
---

[8]

Examiner Only	
Marks	Remark

- 6 YogaLife is a centre offering Yoga from beginner to trainer level. Add-ons such as consultations or meditations are also available. Apart from one-off taster sessions, clients can prepay for courses of 4, 6 or 8 sessions. A Retreat is a course held at an off-site venue.

**Inheritance Diagram for YogaLife**



Examiner Only	
Marks	Remark



(a) Identify which inheritance term best describes the associations for the following:

(i) The class **Course** to the class **Session**;

\_\_\_\_\_ [1]

(ii) A method **Cost()** in the class **Course** to a method **Cost()** in **Session**. (The method **Cost()** in the class **Session** returns the unit cost of a session based on the session level);

\_\_\_\_\_ [1]

(iii) An array of type **Session** also holding courses and retreats.

\_\_\_\_\_ [1]

(b) Should the term **abstract** be applied to the class **Session**?

Underline your choice.

**Yes / No**

[1]

Give a reason for your choice.

\_\_\_\_\_

\_\_\_\_\_ [1]

Examiner Only	
Marks	Remark





Permission to reproduce all copyright material has been applied for.  
In some cases, efforts to contact copyright holders may have been unsuccessful and CCEA  
will be happy to rectify any omissions of acknowledgement in future if notified.