Software Development:
Object Oriented Programming: An Introduction

HIGHER NATIONAL DIPLOMA

软件开发：面向对象编程（初级）

【英】苏格兰学历管理委员会（SQA）

Unit Student Guide

COMPUTING: Software Development

DH3C 35

中国时代经济出版社

SCOTTISH
QUALIFICATIONS
AUTHORITY

Software Development:
Object Oriented Programming: An Introduction

HIGHER NATIONAL DIPLOMA

软件开发：面向对象编程（初级）

【英】苏格兰学历管理委员会（SQA）

Unit Student Guide

COMPUTING: Software Development

DH35

◆ 中国时代经济出版社

SCOTTISH
QUALIFICATIONS
AUTHORITY

Software Development: Object Oriented Programming: An Introduction

软件开发：面向对象编程（初级）

苏格兰学历管理委员会著

Contents

1 Introduction to the Scottish Qualifications Authority

This Unit **DH3C 35 Software Development: Object Oriented Programming** has been devised and developed by the Scottish Qualifications Authority (SQA). Here is an explanation of the SQA and its work:

The SQA is the national body in Scotland responsible for the development, accreditation, assessment, and certification of qualifications other than degrees.

Its website can be viewed on: www.sqa.org.uk

SQA's functions are to:

- devise, develop and validate qualifications, and keep them under review

- accredit qualifications

- approve education and training establishments as being suitable for entering people for these qualifications

- arrange for, assist in, and carry out, the assessment

of people taking SQA qualifications

- quality assure education and training establishments which offer SQA qualifications

- issue certificates to candidates.

In order to pass SQA units, students must complete prescribed assessments. These assessments must meet certain standards.

The Unit Specification outlines the four Outcomes that students must complete in order to achieve this unit. The Specification also details the knowledge and/or skills required to achieve the outcome or outcomes. The Evidence Requirements prescribe the type, standard and amount of evidence required for each outcome or outcomes.

2 Introduction to the Unit

2.1 What is the Purpose of this Unit?

This Unit is designed to develop a broad knowledge of the concepts, principles, boundaries and scope of software development using an object oriented programming language. These will be reinforced by developing the practical skills required to use the structures and features of an object oriented programming language in the creation of software solutions to problems.

2.2 What are the Outcomes of this Unit?

1. Use programming techniques to develop program modules

2. Implement a solution from design

3. Test the completed product

4. Create technical and user documentation.

Further details can be found in Appendix 1 — Unit Specifications.

| 2.3 What do I Need to be Able to do in Order to Achieve this Unit? | You will be required to develop and test an object oriented system and document technical and user guides to accompany this. Although largely practical, there is a closed book assessment for this Unit so you must have a good understanding of object oriented terms and techniques. |

| 2.4 Approximate Study Time for This Unit | Completion of this Unit is intended to be flexible.

The notional study time for this Unit is 80 hours but actual time allocated is at the discretion of the centre. |

| 2.5 Equipment/ Material Required for this Unit | You will need access to a computer running the Java Development Kit (J2SDK). This software is free and can be downloaded from http://java.sun.com. This same site also gives access to the online documentation (API Specifications) and help files for the Java language. |

It is possible to develop code using a plain text editor only, such as Microsoft Window's Notepad. However, this task becomes easier with the use of a text editor associated with Java. There are many packages available including Boreland's JBuilder (freeware for educational use), TextPad (shareware) and JavaIDE (freeware).

You will need to complete the following Unit Student

Guides to meet all the requirements of this unit:

- Unit Student Guide — Software Development: Object Oriented Programming: An Introduction

- Unit Student Guide — Software Development: Object Oriented Programming: Advanced.

2.6 Symbols Used in this Unit

The various Learning Materials sections are designed so that you can work at your own pace, with tutor support. As you work through the Learning Materials (see Section 5), you will encounter symbols. These symbols indicate that you are expected to complete a task. **These tasks are not Outcome Assessments.** They are exercises designed to consolidate learning or encourage thought, in preparation for the Outcome Assessment (see Section 3 — Assessment Information for this Unit).

Activity

1

This symbol indicates an Activity. Usually, Activities are used to improve or consolidate your understanding of the subject in general or a particular feature of it.

In this Unit, you are asked to undertake activities that allow you to practice writing and testing code. These should get you into the habit of good programming practice and develop an awareness of how to code

with an object oriented language.

The code listings for all programs in each Activity are shown at the end of Section 5.2. If you do not have electronic access to these files, this will allow you to re-create the files yourself.

Self-Assessed Question
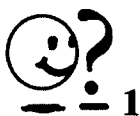
This symbol indicates a Self-Assessed Question. Using a Self-Assessed Question helps you check your understanding of the content that you have already covered. The Self-Assessed Questions in this guide will often take the form of short response questions, multiple choice or true/false questions.

Everything is provided for you to check your own responses. Answers to the Self-Assessed Questions and Activities are to be found at the back of the Unit Student Guide. Where suggested responses to Self-Assessed Question and Activities are provided in the Unit Student Guide, **students are strongly discouraged from looking at these responses before they attempt the activity or question.** The Self-Assessed Questions and Activities throughout the Unit Student Guide will help you to prepare yourself for the formal assessments, and to identify topic areas in which you will require clarification and additional tutor support. The Self-Assessed Questions and Activities

will not serve this purpose if you look at the answers before trying them!

Self-Assessed Questions and Activities are designed to be checked by you. No tutor input is necessary at this stage unless special help is requested, although from time to time your tutor may wish to view your responses to Self-Assessed Questions to see how you are progressing.

3

Assessment Information for this Unit

<table>
<tr><td>

3.1

What Do I Have to Do to Achieve This Unit?

</td><td>

The Unit is assessed by means of a single project covering all four Outcomes with the exception of part of Outcome 2 that is closed book. Although any object oriented language could be used to meet these criteria, this guide is written around the Java programming language.

Further details can be found in Appendix 1 — Unit Specifications.

</td></tr>
</table>

4 Suggested Lesson Plan

The Learning Materials (see Section 5) are designed to lead you through a series of Activities that will allow you to consolidate your learning and check on your own progress.

Activities are given throughout the Unit, at least at the end of each topic. Some are progressive so they should be completed in sequence.