软件开发（事件驱动式高级编程）

COMPUTING: Software Development

HIGHER NATIONAL DIPLOMA

【英】苏格兰学历管理委员会（SQA）
Scottish Qualifications Authority

Unit Student Guide

Software Development: Event Driven Programming: Advanced   DG8V 04

中国时代经济出版社

SCOTTISH
QUALIFICATIONS
AUTHORITY

软件开发（事件驱动式高级编程）

COMPUTING:Software Development

HIGHER NATIONAL DIPLOMA

【英】苏格兰学历管理委员会 (SQA)
Scottish Qualifications Authority

Unit Student's Guide

Software Development:Event Driven Programming: Advanced DG6L 04

中国时代经济出版社

SCOTTISH
QUALIFICATIONS
AUTHORITY

Software Development：
Event Driven Programming: Advanced

软件开发：事件驱动式高级编程

苏格兰学历管理委员会著

Contents

1 Introduction to the Scottish Qualifications Authority

This Unit DG8V 04 Software Development: Event Driven 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 4 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

The purpose of this guide is to assist you to study and then pass the assessments necessary to achieve this HN Unit in computer programming. In so doing you will discover and employ the power of a particular event-driven language: namely, Microsoft Visual Basic, or "VB" for short.

VB is a very popular environment for developing programs. It is a modern and more sophisticated version of BASIC ("Beginner's All purpose Symbolic Instruction Code"), a very early language designed to help people learn to program as long ago as 1963.

There are a number of different versions of VB. These notes are based upon VB 6.0 (Enterprise Edition), but other versions of VB 6 and later will behave in a very similar way. The bulk of the guide involves very basic programming concepts and all the versions of VB after 6.0 can be expected to support the activities taught within this guide.

In order to be efficient, we need to make some assumptions about you, our students, and in the next

section is a list of computer skills and knowledge which you will need, before you can learn from this guide. These are skills and knowledge which will be well within the competence of the average computer user. If you are not already competent at these, then you will need some help before you attempt to start working through this guide. Most colleges run courses such as "Introduction to the Computer", or "Early Steps in Microsoft Windows" or similar. Your college tutor will be able to advise you.

We will also assume that VB is loaded onto the PC which you intend to use, and so will not deal with the installation of VB onto your computer.

In the next section we will look at the requirements of the HN Unit : Software Development : Event Driven Programming as published by the Scottish Qualifications Authority (SQA) . The SQA is the body which sets standards, supervises teaching and assessment, and which will award you your qualification-provided of course, that you pass the various assessments.
We hope very much that you will succeed.

2.1 What is the Purpose of this Unit?

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

solutions to problems, based upon designs produced either by the student or by others.

2. 2
What are the Outcomes of this Unit?

On completion of this Unit, you should be able to:

1. **Use programming techniques to develop program modules.** This will involve you in the use of various controls, and methods of entering, holding and displaying pieces of data like names and numbers.

2. **Implement a solution from a design.** This will require you to interpret the requirements of a given design and build modules of programming code which while performing separate tasks, can be coordinated to achieve the ultimate purpose of the program.

3. **Test the completed product.** This is where you enter appropriate data and establish whether the program meets its specification and design requirements. Most programmers spend a considerable amount of time doing this, and then correcting any errors which are revealed.

4. **Create technical and user documentation.** The purpose of documentation is to enable users to make best use of the software and to enable technical experts to debug, or otherwise maintain the program. It is absolutely essential that commercial software is properly documented and so it is necessary that you learn how to do this.

2.3 What do I Need to be Able to do in Order to Achieve this Unit?

Basic Skills and Knowledge

You should be confident about the following hardware issues:

a) turning on the computer

b) operating a standard keyboard

c) operating a mouse or other pointing device, using left, right or centre button as required

d) inserting a CD or diskette into the respective disk drive, ready to be used.

You should be confident about the following software / hardware skills:

- "logging in" to your PC, as appropriate

- launching the VB software program

- clicking control buttons

- causing drop down menus to be displayed

- selecting and activating an option from a drop down menu

- dragging toolbars to a chosen position and orientation

- moving display panes to suitable positions within the VB window.

- sizing display panes to your own requirements

- closing display panes by use of the " × " button in the top right corner

- causing scrollbars to move-either by dragging or by clicking

- selecting one from amongst several option buttons.

- causing a tick mark （√） to appear in or disappear from check boxes

- selecting portions of text （resulting in the text being highlighted for processing） using the mouse or the keyboard

- copy, cut and paste portions of text as required.

You should work through this learner guide attempting all activities and self assessed questions to confirm your own understanding of each topic.

Tutor-marked assessments should be submitted as directed by the class lecturer.

You will be required to complete 2 Student Unit Guides to achieve the unit:

Software Development Event Driven Programming: An

Introduction

Software Development Event Driven Programming:
   Advanced

2. 4
Approximate
Study Time for
This Unit?

The average time allowed the complete the unit is 80 hours; however, you may take less time or you may take longer. Try to follow the schedule provided by your tutor.

2. 5
Equipment/
Material
Required   for
this Unit

Suitable PC (capable of running Microsoft Visual Basic 6. 0)

Loaded software to run :
   VB6
   VB Documentation from MSDN Library from Visual Studio 6

Access to Student Guide Software comprising:
   Demonstration Code
   Problem Code
   Problem Solution Code
   Demonstration exe programs
(This may be by means of CD or network access)

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 do 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 （A）. Such activities are used to develop, improve or consolidate your understanding of the subject in general or a particular feature of it.

In this unit, you are asked to undertake the writing and sometimes correcting of computer programs. In some sections of the Guide, this work is developmental and you will find demonstration solution programs which achieve each stage.

The activities will not serve this purpose if you refer to the solutions prior to having attempted the Activity. However, when the text suggests it, we strongly urge you to run demonstration **(exe)** programs, in order that you can see the purpose or the methods that the work is intended to achieve.

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. There are few of these because the bulk of the self-assessment activity is practical.

Everything is provided for you to check your own responses. Answers to the Self Assessed Questions are to be found at the back of the Unit Student Guide. Where suggested responses to activities are provided in the Unit Student Guide, **students are strongly discouraged from looking at these responses before they attempt the activity.** The 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 activities will not serve this purpose if you look at the answers before trying the activity!

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.

2.7 Terminology Used in this Unit

Term	Definition
code (program code)	the collective term for a number of written statements which can be carried out, or executed, as a running program.
VB Integrated Environment	A software interface through which you can write and save program code and then can test run it to see if it performs correctly.
Design time	The period of time when your program is being developed by writing and editing code statements within the VB Integrated Environment.
Run time	The period of time when your program (or application) is running. At run time, the programmer interacts with the application as the user would.
Controls	These are the devices such as buttons, list boxes, scroll bars, menus and text boxes, which users can use to provide the information which needs to be input into your program or which can display the output.
Forms	Windows that can contain controls for user input and for program output. At least one form will serve as the interface by which the user can enter data or read the output from your program.
Objects	A general term used to describe all the forms and controls that make up a program. Each object has a number of properties (typically 15 or 20 such properties) which record its location and size and other aspects too.

Term	Definition
Properties	The technical term for the characteristics of an object such as size, caption, or colour. In general you can change the appearance of an object by changing its properties. In general, this can be done at design time or at run time.
Methods	The actions which an object can itself perform, or which can be performed on the object in question. For example, a progress bar typically shows what percentage of time needed for a particular activity has passed. This is such a method. On the other hand, you could move a scroll bar in order to input a particular value. This is also a method. Most objects have at least a dozen such methods available for you to use.
Events	Occurrences which are detected by the objects in your program. Forms and controls are able to recognise mouse clicks, mouse location, text and other keys, the passing of time and much more. Events are typically triggered by the user, the operating system, other programs or other parts of your program.
procedure	A sequence of code statements which are executed as a block in order to carry out a particular task. Some procedures carry out calculations and others compare values or print out results. Each procedure has a name so that it can be called (activated).