Australian Government
Department of Education, Employment and Workplace Relations

ICAB5223B Apply intermediate object-oriented language skills

Release: 1

ISC INDUSTRY SKILLS COUNCILS
Creating Australia's Future

ICAB5223B Apply intermediate object-oriented language skills

Modification History

Not Applicable

Unit Descriptor

Unit descriptor	This unit defines the competency required to undertake intermediate level programming tasks using an object-oriented programming language. No licensing, legislative, regulatory or certification requirements apply to this unit at the time of publication.

Application of the Unit

Application of the unit	

Licensing/Regulatory Information

Refer to Unit Descriptor

Pre-Requisites

Prerequisite units		
	ICAB4219B	Apply introductory object-oriented language skills

Employability Skills Information

Employability skills	This unit contains employability skills.

Elements and Performance Criteria Pre-Content

Elements describe the essential outcomes of a unit of competency.	Performance criteria describe the performance needed to demonstrate achievement of the element. Where bold italicised text is used, further information is detailed in the required skills and knowledge section and the range statement. Assessment of performance is to be consistent with the evidence guide.

Elements and Performance Criteria

ELEMENT	PERFORMANCE CRITERIA
1. Build applications using provided language utilities	1.1. Divide multiple source code files into logical units/packages 1.2. Use at least two of the utilities of the target language allowing for internal storage of ***collections of data*** 1.3. Use the utilities of the target language providing internal data sorting and searching facilities 1.4. Employ ***integrated development environment*** facilities to make files to automate program building 1.5. Follow guidelines for developing maintainable code adhering to ***coding standards*** 1.6. Use the facilities in the language for persisting objects to binary files 1.7. Use the operator and function/method overloading facilities available in the ***language*** at an introductory level 1.8. Demonstrate ability to use exception handling techniques to ensure program stability 1.9. Demonstrate use of a class that is based on multiple inheritances
2. Write programs that interact with a database	2.1. Design and implement programs that connect to a ***database*** 2.2. Design and implement programs that use the languages facilities to extract, update and delete data stored in a database 2.3. Design and implement programs that use the languages facilities to manipulate database structure (query, create and delete) 2.4. Write programs that deliver transactional integrity
3. Write GUI	3.1. Employ GUI framework or text windowing interface appropriate to the chosen language 3.2. Demonstrate use of standard ***GUI components*** 3.3. Use the facilities within the language for GUI objects to respond to user and program generated events
4. Debug application	4.1. Use standalone debugging tools or tools provided by ***integrated development environment*** to examine variables and trace running code 4.2. Use debugger to detect logical and coding errors 4.3. Use tracing of code and examination of variable

ELEMENT	PERFORMANCE CRITERIA
	contents during execution to detect and correct errors
5. Test application	5.1. Design and document limited tests of code 5.2. Undertake limited testing of produced code to ensure program specification is complied with 5.3. Capture and document test results
6. Create and maintain documentation	6.1. Read and interpret supplied design document to create code 6.2. Create and maintain program ***documentation***

Required Skills and Knowledge

REQUIRED SKILLS AND KNOWLEDGE
This section describes the skills and knowledge required for this unit.
Required skills
• Interpreting program specifications • Translating requirements from problem space to machine space • Integrated development environment usage • Programming techniques • Internal (code) documentation techniques • Debugging techniques • Testing techniques • Documentation techniques • Reporting and bug tracking
Required knowledge
• Object-oriented programming concepts • Small size application development • Using a GUI to interact with operator • Object-oriented programming language • Medium size application development • Data structures • Documentation techniques

Evidence Guide

EVIDENCE GUIDE	
The evidence guide provides advice on assessment and must be read in conjunction with the performance criteria, required skills and knowledge, range statement and the Assessment Guidelines for the Training Package.	
Overview of assessment	
Critical aspects for assessment and evidence required to demonstrate competency in this unit	Evidence of the following is essential: • Assessment must confirm that application programs are designed and built from a problem scenario and program specification. • Evidence for this competency includes tool usage, documentation, debugging and testing techniques in support of the programming activities and includes database and files. • Design and code documentation must be generated. Testing must confirm that created application meets original specification and solves original problem.
Context of and specific resources for assessment	Programming in object-oriented languages is a software development methodology that offers the programmer standard reusable software modules (components), rather than requiring the developer to write custom programming code each time. Using standard components reduces development time (because the writing and testing of those components has already been done by other programmers), and ensures a standard look and feel for programs using the same components. Object-oriented languages are an important feature of software development processes world-wide. The breadth, depth and complexity covering planning and initiation of alternative approaches to skills or knowledge applications across a broad range of technical and/or management requirements, evaluation and coordination would be characteristic. Assessment must ensure: • The demonstration of competency may also require self-directed application of knowledge and skills, with substantial depth in some areas where judgement is required in planning and selecting appropriate equipment, services and techniques for

EVIDENCE GUIDE	
	self and others. • Applications involve participation in development of strategic initiatives as well as personal responsibility and autonomy in performing complex technical operations or organising others. It may include participation in teams including teams concerned with planning and evaluation functions. Group or team coordination may also be involved.
Method of assessment	The purpose of this unit is to define the standard of performance to be achieved in the workplace. In undertaking training and assessment activities related to this unit, consideration should be given to the implementation of appropriate diversity and accessibility practices in order to accommodate people who may have special needs. Additional guidance on these and related matters is provided in ICA05 Section 1. • Competency in this unit should be assessed using summative assessment to ensure consistency of performance in a range of contexts. This unit can be assessed either in the workplace or in a simulated environment. However, simulated activities must closely reflect the workplace to enable full demonstration of competency. • Assessment will usually include observation of real or simulated work processes and procedures and/or performance in a project context as well as questioning on underpinning knowledge and skills. The questioning of team members, supervisors, subordinates, peers and clients where appropriate may provide valuable input to the assessment process. The interdependence of units for assessment purposes may vary with the particular project or scenario.
Guidance information for assessment	Holistic assessment with other units relevant to the industry sector, workplace and job role is recommended. An individual demonstrating this competency would be able to:

EVIDENCE GUIDE	
	• Demonstrate understanding of a broad knowledge base incorporating theoretical concepts, with substantial depth in some areas • Analyse and plan approaches to technical problems or management requirements • Transfer and apply theoretical concepts and/or technical or creative skills to a range of situations • Evaluate information, using it to forecast for planning or research purposes • Take responsibility for own outputs in relation to broad quantity and quality parameters • Take some responsibility for the achievement of group outcomes • Maintain knowledge of industry products and services

Range Statement

RANGE STATEMENT	
The range statement relates to the unit of competency as a whole. It allows for different work environments and situations that may affect performance. Bold italicised wording, if used in the performance criteria, is detailed below. Essential operating conditions that may be present with training and assessment (depending on the work situation, needs of the candidate, accessibility of the item, and local industry and regional contexts) may also be included.	
Integrated development environment may include but is not limited to:	• Visual C++ • Visual Studio suite • Eclipse • J-Edit • Code Warrior
Collections of data may include but is not limited to:	• lists • trees • hash tables • sets • stacks • queues
Coding standards may include:	• Java coding standard • GNU coding standard

RANGE STATEMENT	
Language may include but is not limited to:	- Java - C++ - Small Talk - VB.net - C#.net
Database may be:	- relational - object
GUI components may include:	- buttons - check boxes - option buttons - drop-down lists - text input fields, etc.
Documentation may include but is not limited to:	- user manuals - design documents - requirement documents - test documents - release documents - in-code documentation - code comments - internal module documentation - architecture documentation - design document

Unit Sector(s)

Unit sector	Build

Co-requisite units

Co-requisite units		

Competency field

Competency field	