VCE Applied Computing 2025–2028

Video 3

Background to Unit 1 Outcome 2
Applied Computing

VICTORIAN CURRICULUM AND ASSESSMENT AUTHORITY
VICTORIA State Government

Acknowledgement of Country

The VCAA respectfully acknowledges the Traditional Owners of Country throughout Victoria and pays respect to the ongoing living cultures of First Peoples.

VCE Applied Computing 2025–2028

Video 3

Background to Unit 1 Outcome 2
Applied Computing

VICTORIAN CURRICULUM AND ASSESSMENT AUTHORITY
VICTORIA State Government

Purpose of this presentation

- Overview of Unit 1 Outcome 2 Applied Computing
- Major changes to Unit 1 Outcome 2
- Software tools
- Outcome statement
- Key knowledge
- Key skills
- Assessment tasks

Unit 1 Outcome 2

Changes to Unit 1 Outcome 2

- Emerging trends
- Object-oriented Programming
- Removal of project management
- Updated assessment tasks

Unit 1 Outcome 2

Software tools

Students are required to use the following software tool:

- An appropriate object-oriented programming language

Unit 1 Outcome 2

From the outcome statement

- Interpret teacher-provided solution requirements to design and develop a software solution using an object-oriented programming language.

Unit 1 Outcome 2 – Key knowledge

- emerging trends in programming languages and artificial intelligence-based (AI) code generators for the development of software solutions, such as:
  - low-code development approaches
  - readability and/or maintainability improvements
- characteristics of data types, such as:
  - text (character, string)
  - numeric (integer, floating point, date/time)
  - Boolean

Unit 1 Outcome 2 – Key knowledge

- types of data structures, such as:
  - one-dimensional arrays
  - lists
  - records (varying data types, field index)
- characteristics of functional and non-functional requirements, constraints and scope
- design tools for representing the functionality and appearance of solution designs, such as:
  - mock-ups
  - input-process-output (IPO) charts
  - flowcharts/pseudocode

Unit 1 Outcome 2 – Key knowledge

- key legal requirements relating to intellectual property and copyright while designing and developing software
- principles of OOP, such as:
  - abstraction
  - encapsulation
- features of a programming language, such as:
  - variables, and initialising, accessing and storing data in variables
  - control structures (sequence, selection and iteration/repetition)
  - arithmetic, logical and conditional operators
  - procedures, functions and methods

Unit 1 Outcome 2 – Key knowledge

- naming conventions for solution elements, such as:
  - Hungarian notation
  - camel casing
- purposes of internal documentation, such as:
  - explaining data and code structures
  - code maintenance
- validation techniques for data, such as:
  - existence checking
  - type checking
  - range checking

Unit 1 Outcome 2 – Key knowledge

- debugging and testing techniques for checking software solutions function correctly, such as:
  - test tables to compare expected and actual output
  - construction of relevant test data
  - breakpoints
  - debugging output statements.

Unit 1 Outcome 2 – Key skills

- interpret solution requirements to develop a software solution
- select and use appropriate design tools to represent solution designs
- use a range of data types and data structures
- develop a software solution using appropriate features of an OOP language
- document the functioning of a software solution through internal documentation
- design and apply suitable validation, debugging and testing techniques.

Unit 1 Outcome 2 – Assessment task

Suitable tasks for assessment in this unit may be selected from the following:

- A folio of exercises to demonstrate the learning of an object-oriented programming language.
- A software solution that includes the designs, solution and a testing table in response to teacher-provided solution requirements.
- The creation and maintenance of code repositories to track the progression of students' learning, using platforms such as GitHub.
- A software solution developed in response to a teacher-provided problem-solving challenge, presented as a hackathon.

Contact

- Phil Feain – Digital Technologies Curriculum Manager (VCAA)
- Ph: (03) 9059 5146
- Philip.Feain@education.vic.gov.au

Authorised and published by the
Victorian Curriculum and Assessment Authority