Australian Government

Department of Education, Employment and Workplace Relations

ICAPRG428A Use regular expressions in programming languages

Release: 1

INDUSTRY SKILLS COUNCILS

Creating Australia's Future

ICAPRG428A Use regular expressions in programming languages

Modification History

Release	**Comments**
Release 1	This Unit first released with *ICA11 Information and Communications Technology Training Package version 1.0*

Unit Descriptor

This unit describes the performance outcomes, skills and knowledge required to use regular expressions for manipulating text and data.

Application of the Unit

Programmers and system administrators apply the skills and knowledge described in this unit. Their main responsibility is to code complex text processing such as search and replace tasks, validate inputs and create regex-based queries.

Licensing/Regulatory Information

No licensing, legislative, regulatory or certification requirements apply to this unit at the time of endorsement but users should confirm requirements with the relevant federal, state or territory authority.

Pre-Requisites

Not applicable.

Employability Skills Information

This unit contains employability skills.

Elements and Performance Criteria Pre-Content

Element	**Performance Criteria**
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

1. Differentiate the use of regular expressions in programming languages	1.1 Compare regular expression features among different ***languages*** and tools 1.2 Define different ***regular expression engine*** types and how they work 1.3 Determine the programming language and tools for writing regular expressions
2. Code using regular expressions	2.1 Use regular expression characters, operators, anchors and quantifiers to match patterns of text 2.2 Write regular expression ***patterns*** to search and manipulate text 2.3 Write code to split strings and collect matches 2.4 Validate user input using regular expressions
3. Test regular expressions	3.1 Use ***test tools*** to test and validate regular expressions 3.2 Capture results and edit as required

Required Skills and Knowledge

This section describes the skills and knowledge required for this unit.

Required skills

- analytical skills to:
  - evaluate and compare a range of tools and languages
  - write complex regular expressions for text processing
- literacy skills to read basic technical data
- problem-solving skills to:
  - find best solutions using regular expressions to a wide range of problems
  - perform basic debugging and testing skills
- research skills to locate and interrogate complex and varied sources of information
- technical skills to:
  - debug applications using basic debugging techniques
  - test applications using basic testing techniques
  - write code using an integrated development environment (IDE).

Required knowledge

- maths at a basic level
- basic knowledge of:
  - coding
  - database systems
  - object-oriented programming
  - open-source development tools
  - small-size application development.

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 ability to: • build and test regular expressions • match, replace or split text within a document or input field.
Context of and specific resources for assessment	Assessment must ensure access to: • internet in order to use online tools available for regular expressions • IDE • specific tools and licences, depending on particular platform • appropriate learning and assessment support when required • modified equipment for people with special needs.
Method of assessment	A range of assessment methods should be used to assess practical skills and knowledge. The following examples are appropriate for this unit: • verbal or written questioning to assess knowledge of writing regular expressions to solve common text-processing problems • evaluation of ability to: • create code to search files and input fields • handle data validation • manipulate text based on patterns.
Guidance information for assessment	Holistic assessment with other units relevant to the industry sector, workplace and job role is recommended, where appropriate. Assessment processes and techniques must be culturally appropriate, and suitable to the communication skill level, language, literacy and numeracy capacity of the candidate and the work being performed. Indigenous people and other people from a non-English speaking background may need additional support. In cases where practical assessment is used it should be combined with targeted questioning to assess required knowledge.

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.

Languages may include:	• .NET languages • C++ • Java • JavaScript • MySQL • Perl • PHP • PowerShell • Python • Ruby on Rails • XML Schema • XQuery and XPath.
Regular expression engine types may include:	• deterministic finite automaton (DFA) • non-deterministic finite automaton (NFA) • portable operating system interface for Unix NFA (POSIX NFA).
Patterns may include:	• alternatives • character classes • grouping • repeating sequences • sub-patterns.
Test tools may include:	• Grep • java.util.regex • PowerGREP • RegexBuddy • RegexMagic • RegexPal • Regular Expression Checker • Rubular.

Unit Sector(s)

Programming and software development