



Australian Government

ICTPRG428 Use regular expressions in programming languages

Release: 1

ICTPRG428 Use regular expressions in programming languages

Modification History

Release	Comments
Release 1	This version first released with ICT Information and Communications Technology Training Package Version 1.0.

Application

This unit describes the skills and knowledge required to use regular expressions (regex) for manipulating text and data.

It applies to individuals who are programmers and system administrators, and whose main responsibilities are to code complex text processing, such as search-and-replace tasks, validate inputs, and create regex-based queries.

No licensing, legislative or certification requirements apply to this unit at the time of publication.

Unit Sector

Programming and software development

Elements and Performance Criteria

ELEMENT	PERFORMANCE CRITERIA
<i>Elements describe the essential outcomes.</i>	<i>Performance criteria describe the performance needed to demonstrate achievement of the element.</i>
1. Differentiate the use of regular expressions in programming languages	1.1 Compare the regular expression features among the different languages and tools 1.2 Define the 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 the patterns of text 2.2 Write regular expression patterns to search, and manipulate,

ELEMENT	PERFORMANCE CRITERIA
	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

Foundation Skills

This section describes language, literacy, numeracy and employment skills incorporated in the performance criteria that are required for competent performance.

Skill	Performance Criteria	Description
Reading	1.1	Interprets and critically analyses complex texts, and applies appropriate strategies, to construct the meaning from complex texts, when comparing regex features in different languages and tools
Writing	1.2, 2.1, 2.2, 2.3, 3.2	- Communicates complex relationships between ideas and information, matching the style of writing to the purpose and audience - Writes, and edits, computer code and technical data, ensuring the correct syntax and accuracy
Numeracy	2.1, 2.2, 2.3, 2.4	Selects from, and applies, an expanding range of mathematical and problem-solving strategies
Get the work done	1.1, 1.2, 1.3, 2.1, 2.2, 2.3, 2.4, 3.1, 3.2	- Uses systematic, analytical processes in complex, non-routine situations, setting goals, gathering relevant information, and identifying, and evaluating options against the agreed criteria - Uses analytical processes to decide on a course of action, establishing the criteria for deciding between options when evaluating, and comparing, tools and languages, and when using and testing regex - Understands the key principles and concepts underpinning the design, and operation, of digital systems and tools

Unit Mapping Information

Code and title current version	Code and title previous version	Comments	Equivalence status
ICTPRG428 Use regular expressions in programming languages	ICAPRG428A Use regular expressions in programming languages	Updated to meet Standards for Training Packages	Equivalent unit

Links

Companion Volume implementation guides are found in VETNet -

<https://vetnet.gov.au/Pages/TrainingDocs.aspx?q=a53af4e4-b400-484e-b778-71c9e9d6aff2>

Assessment Requirements for ICTPRG428 Use regular expressions in programming languages

Modification History

Release	Comments
Release 1	This version first released with ICT Information and Communications Technology Training Package Version 1.0.

Performance Evidence

Evidence of the ability to:

- compare regular expression (regex) languages, tools and engine types
- prepare and validate code using regex
- test and edit regex code.

Note: If a specific volume or frequency is not stated, then evidence must be provided at least once.

Knowledge Evidence

To complete the unit requirements safely and effectively, the individual must:

- describe and explain the principles of:
 - coding
 - database systems
 - object-oriented programming
 - open-source development tools
 - small-size application development.

Assessment Conditions

Gather evidence to demonstrate consistent performance in conditions that are safe and replicate the workplace. Noise levels, production flow, interruptions and time variances must be typical of those experienced in the programming and software development industry, and include access to:

- the internet, in order to use the online tools available for regular expressions
- an integrated development environment (IDE)
- specific tools and licences, depending on the particular platform.

Assessors must satisfy NVR/AQTF assessor requirements.

Links

Companion Volume implementation guides are found in VETNet -

<https://vetnet.gov.au/Pages/TrainingDocs.aspx?q=a53af4e4-b400-484e-b778-71c9e9d6aff2>