

" LPI® Open Source Essentials (050-100)"

LPI Open Source Essentials - Course Introduction

Open Source Essentials is an education and certificate program that covers the fundamental aspects of free and open source software, open content and open technology. It includes topics from various professions, including legal, technical and business aspects. The program is not limited to a specific profession. Instead, it covers the common knowledge that anyone who is professionally involved with open source should have. This common knowledge allows developers, managers, lawyers and any other professional working in the field of open source to communicate, share a common perspective and terminology, and collaborate successfully. This program also allows individuals to understand the concepts behind open source and engage with the open source community.

051 Software Fundamentals

051.1 Software Components (weight: 2)

Weight	2
Description	<i>The candidate should understand the main characteristics of software. This includes the distinction between hardware and software, the role of source code, as well as the concepts of libraries, static and dynamic linking, and compiling and interpreting source code.</i>

Key Knowledge Areas:

- *Understanding the concept of source code and code execution*
- *Understanding the concept of compilers and interpreters*
- *Understanding the concept of software libraries*

Partial list of the used files, terms, and utilities:

- *Source code*
- *Executable programs*
- *Byte code*

- *Machine code*
- *Compiler*
- *Linker*
- *Interpreter*
- *Runtime virtual machine*
- *Algorithm*
- *Software libraries*
- *Static and dynamic linking*

051.2 Software Architecture (weight: 2)

Weight	2
Description	<i>The candidate should understand the main aspects of common software architectures. This includes understanding the concept of client and server computing, thin and fat clients as well as the role of web applications, single-page applications and application programming interfaces.</i>

Key Knowledge Areas:

- *Understanding the concepts of client and server computing*
- *Understanding the concepts of thin and fat clients*
- *Understanding the concepts of monoliths and microservices and their main differences*
- *Understanding the concepts of Application Programming Interfaces (APIs)*
- *Understanding the concept of software components and their integration or separation (services, modules, APIs)*

Partial list of the used files, terms, and utilities:

- *Clients and servers*
- *Thin clients and fat clients*
- *Web applications*

- *Single-page applications*
- *Monolithic architectures*
- *Microservice architectures*
- *Application Programming Interfaces (APIs)*
- *RESTful APIs*

051.3 On-Premises and Cloud Computing (weight: 1)

Weight	1
Description	<i>The candidate should understand the main options for running software. This includes the distinction between software run on-premise and software run remotely, as well as the various kinds of cloud computing.</i>

Key Knowledge Areas:

- *Understanding the concepts of on-premise and cloud computing*
- *Understanding common cloud operation models*
- *Understanding common types of cloud services*
- *Understanding the major benefits and risks of cloud computing and on-premise IT infrastructure*

Partial list of the used files, terms, and utilities:

- *Cloud computing*
- *On-premises IT infrastructure*
- *Data center*
- *Public, private and hybrid cloud*
- *Infrastructure as a Service (IaaS), Platform as a Service (PaaS), Software as a Service (SaaS)*
- *Cost models*
- *Security*

- *Data ownership*
- *Service availability*

052 Open Source Software Licenses

052.1 Concepts of Open Source Software Licenses (weight: 3)

Weight	2
Description	<i>The candidate should understand the principle and philosophy underlying open source software. This includes understanding the concepts of software licenses, as well as the history of open source and the various usages of software that may be affected by different software licenses.</i>

Key Knowledge Areas:

- *Understanding the definitions of open source software and free software*
- *Awareness of other kinds of monetarily-free software*
- *Awareness of important events in the history of open source*
- *Understanding what a license is and what rights licenses commonly manage*
- *Understanding how existing software can be used to create derivative works*
- *Understanding license compatibilities and incompatibilities*
- *Understanding dual licensing and conditional licensing*
- *Understanding consequences of license violations*
- *Understanding the principles of copyright law and patent law and how they are affected by open source software licenses*

Partial list of the used files, terms, and utilities:

- *Free Software Foundation (FSF) free software definition*
- *Open Source Initiative (OSI) open source software definition*
- *Licenses*
- *Contracts*

- *Public domain software*
- *Freeware*
- *Shareware*
- *License stewards*
- *Permissions to use, modify and distribute code and software*
- *Derivative works and code reuse*
- *Closed source / proprietary software*
- *Paid distribution*
- *Modified and unmodified software distribution*
- *Hosting software as a paid service*
- *License compatibility*
- *Dual and multi licensing*
- *Conditional licensing*
- *Software patents*
- *Explicit and implicit patent license grants*

052.2 Copyleft Software Licenses (weight: 3)

Weight	3
Description	<i>The candidate should understand important properties of commonly used copyleft software licenses. This includes understanding the permissions granted and obligations imposed by the licenses, as well as license compatibility.</i>

Key Knowledge Areas:

- *Understanding the concept of copyleft software licenses*
- *Understanding the rights granted by copyleft software licenses*
- *Understanding the obligations created by copyleft software licenses*

- *Understanding the main properties of common copyleft software licenses*
- *Understanding the compatibility of copyleft software licenses with other software licenses*
- *Awareness of the terms 'reciprocal license' and 'restrictive license'*

Partial list of the used files, terms, and utilities:

- *Copyleft*
- *Distributing*
- *Conveying*
- *Tivoisierung*
- *GNU General Public License, version 2.0 (GPLv2)*
- *GNU General Public License, version 3.0 (GPLv3)*
- *GNU Lesser General Public License, Version 2 (LGPLv2)*
- *GNU Lesser General Public License, Version 3 (LGPLv3)*
- *GNU Affero General Public License, Version 3 (AGPLv3)*
- *Eclipse Public License (EPL), version 1.0*
- *Eclipse Public License (EPL), version 2.0*
- *Mozilla Public Licence (MPL)*

052.3 Permissive Software Licenses (weight: 3)

Weight	3
Description	<i>The candidate should understand important properties of commonly used permissive software licenses. This includes understanding the permissions granted and obligations imposed by the licenses, as well as license compatibility.</i>

Key Knowledge Areas:

- *Understanding the concept of permissive software licenses*

- *Understanding the rights granted by permissive software licenses*
- *Understanding the obligations created by permissive software licenses*
- *Understanding the main properties of common permissive software licenses*
- *Understanding the compatibility of permissive software licenses with other licenses*

Partial list of the used files, terms, and utilities:

- *2-Clause BSD License*
- *3-Clause BSD License*
- *MIT License*
- *Apache License, version 2.0*

053 Open Content Licenses

053.1 Concepts of Open Content Licenses (weight: 2)

Weight	2
Description	<i>The candidate should understand important aspects of open content licenses. This includes the various kinds of consent, as well as the principles of existing open content licenses.</i>

Key Knowledge Areas:

- *Understanding types of open content*
- *Understanding what constitutes content that is subject to copyright*
- *Understanding derivative works of copyrighted materials*
- *Understanding the need for open content licenses*
- *Awareness of trademarks*

Partial list of the used files, terms, and utilities:

- *Documentation*
- *Images*

- *Artwork*
- *Maps*
- *Music*
- *Videos*
- *Hardware designs and specifications*
- *Databases*
- *Data streams*
- *Data feeds*

053.2 Creative Commons Licenses (weight: 2)

Weight	2
Description	<i>The candidate should understand important properties of Creative Commons licenses. This includes understanding the permissions granted and obligations imposed by the licenses, as well as license compatibility.</i>

Key Knowledge Areas:

- *Understanding the concept of Creative Commons licenses*
- *Understand the Creative Commons license types and their combinations*
- *Understanding the rights granted by Creative Commons licenses*
- *Understanding the obligations created by Creative Commons licenses*

Partial list of the used files, terms, and utilities:

- *Public Domain Dedication (CC0)*
- *Creative Commons Attribution (CC BY)*
- *Creative Commons Attribution-ShareAlike (CC BY-SA)*
- *Creative Commons Attribution-NonCommercial (CC BY-NC)*
- *Creative Commons Attribution-NonCommercial-ShareAlike (CC BY-NC-SA)*

- *Creative Commons Attribution-NoDerivatives (CC BY-ND)*
- *Creative Commons Attribution-NonCommercial-NoDerivatives (CC BY-NC-ND)*

053.3 Other Open Content Licenses (weight: 1)

Weight	1
Description	<i>The candidate should understand important properties of open content licenses other than Creative Commons. This includes understanding the permissions granted and obligations imposed by the licenses, as well as license compatibility.</i>

Key Knowledge Areas:

- *Understanding licensing for documentation*
- *Understanding licensing for data sets and databases*
- *Understanding the rights granted by open content licenses*
- *Understanding the obligations created by open content licenses*

Partial list of the used files, terms, and utilities:

- *GNU Free Documentation License, version 1.3 (GFDL)*
- *Open Data Commons Open Database License (ODbL)*
- *Community Data License Agreement – Permissive, version 1.0 (CDLA)*
- *Community Data License Agreement – Sharing, version 1.0 (CDLA)*
- *Open Access*

Topic 054 Open Source Business Models (weight: 7)

054.1 Software Development Business Models (weight: 2)

Weight	2
Description	<i>The candidate should understand important aspects of software development</i>

	<i>business models related to open source software.</i>
--	---

Key Knowledge Areas:

- *Understanding goals and reasons to release software or content under an open license*
- *Understanding common business models and revenue streams for organizations developing open source software and open content*
- *Understanding the implications of using open source software as components in larger technology products and services*
- *Understanding the impact licenses have on a software development business models*
- *Understanding considerations of open source software from a client's perspective*
- *Awareness of cost structures and investments needed for open source software development business models*

Partial list of the used files, terms, and utilities:

- *Paid development*
- *Open-core and paid add-ons*
- *Freemium*
- *Enterprise and community versions*
- *Self-hosted distribution*
- *Subscriptions*
- *Customer support*

054.2 Service Provider Business Models (weight: 2)

Weight	2
Description	<i>The candidate should understand important aspects of service provider business models related to open source software.</i>

Key Knowledge Areas:

- *Understanding common business models and revenue streams for organizations*

providing services related to open source software and open content

- *Understanding the impact licenses have on a service provider business models*
- *Understanding service level objectives and service level agreements*
- *Understanding the need for security and privacy protection*
- *Awareness of cost structures and investments needed for open source software service business models*

Partial list of the used files, terms, and utilities:

- *Hosted services*
- *Clouds*
- *Consulting*
- *Training*
- *Hardware sales*
- *User support*
- *Terms of Service (ToS)*
- *Service Level Objectives (SLO)*
- *Service Level Agreements (SLA)*
- *Data processing agreements*

054.3 Compliance and Risk Mitigation (weight: 3)

Weight	3
Description	<i>The candidate should understand important legal aspects of developing and using open source software in a professional environment. This includes license compliance, legal compliance, asset protection, privacy law and aspects related to entrepreneurship.</i>

Key Knowledge Areas:

- *Understanding how to ensure license compliance*

- *Understanding how to maintain information about licenses*
- *Understanding the concept of Open Source Program Offices*
- *Understanding the implications of copyright, patents and trademarks on open source business models*
- *Awareness of legal risks related to open source business models*
- *Awareness of financial risks related to open source business models*

Partial list of the used files, terms, and utilities:

- *Software Composition Analysis (SCA)*
- *Software Bill Of Materials (SBOM)*
- *Software Package Data Exchange (SPDX)*
- *OWASP CycloneDX*
- *Open Source Program Offices (OSPO)*
- *Product warranty*
- *Product liability*
- *Export regulations*
- *Impact of mergers and acquisitions*

055 Project Management

055.1 Software Development Models (weight: 3)

Weight	3
Description	<i>The candidate should understand important aspects of managing an open source software development project. This includes common project management methodologies for software development, including project roles.</i>

Key Knowledge Areas:

- *Understanding the relevance and goals of project management in software development*
- *Basic understanding of waterfall software development*

- *Basic understanding of agile software development, including Scrum and Kanban*
- *Understanding the concept of DevOps*

Partial list of the used files, terms, and utilities:

- *Phases in waterfall projects (requirement engineering, business analysis, software design, development, testing, operations)*
- *Roles in waterfall projects (project managers, business analysts, software architects, developers, testers)*
- *Organization of Scrum projects (sprints and sprint planning, product and sprint backlog, daily scrums, sprint review and sprint retrospective)*
- *Roles in Scrum projects (product owners, developers, scrum masters)*
- *Organization of Kanban projects (Kanban boards)*

055.2 Product Management / Release Management (weight: 2)

Weight	2
Description	<i>The candidate should understand important aspects of managing an open source software project. This includes defining milestones, releases and versioning.</i>

Key Knowledge Areas:

- *Understanding common release types*
- *Understanding software versioning and reasons for major or minor releases*
- *Understanding the lifecycle of a software product, from its planning, development and release to its retirement*
- *Understanding the documentation for product versions*

Partial list of the used files, terms, and utilities:

- *Alpha and beta versions*
- *Release candidates*
- *Feature freeze*

- *Major and minor releases*
- *Semantic versioning*
- *Roadmaps and milestones*
- *Changelogs*
- *Long Term Support (LTS)*
- *End of Life (EOL)*
- *Backward compatibility*

055.3 Community Management (weight: 2)

Weight	2
Description	<i>The candidate should understand important aspects of open source communities. This includes common roles, procedures and documents used to organize open source software development projects.</i>

Key Knowledge Areas:

- *Understanding roles in open source projects*
- *Understanding common tasks in open source projects*
- *Understanding the various kinds of contributions to open source projects*
- *Understanding the various kinds of contributors to open source projects*
- *Understanding the role of organizations in maintaining open source projects*
- *Understanding the transfer of rights from individuals to an organization maintaining a project*
- *Understanding rules and policies in open source projects*
- *Understanding attribution and transparency on contributions*
- *Understanding aspects of diversity, equity, inclusivity and non-discrimination*

Partial list of the used files, terms, and utilities:

- *Software development*

- *Documentation*
- *Designs and artwork*
- *User support*
- *Developers*
- *Release managers*
- *Users*
- *Project leads and benevolent dictators*
- *Individuals and corporations*
- *Enthusiasts and professionals*
- *Core team members and occasional contributors*
- *Code and documentation contributions*
- *Bug reporting*
- *Forks*
- *Foundations and sponsors*
- *Contribution agreements*
- *Developer Certificates of Origin (DCO)*
- *Coding guidelines*
- *Codes of conduct*

056 Collaboration and Communication

056.1 Development Tools (weight: 2)

Weight	2
Description	<i>The candidate should understand the main categories and features of tools commonly used to develop software.</i>

Key Knowledge Areas:

- *Understanding common software development tools*
- *Understanding common deployment environments*
- *Understanding common types of software testing*
- *Understanding the concepts of Continuous Integration and Continuous Delivery (CI/CD)*

Partial list of the used files, terms, and utilities:

- *Integrated Development Environments (IDEs)*
- *Linters*
- *Compilers*
- *Debuggers*
- *Reverse Engineering*
- *Refactoring*
- *Unit testing*
- *Integration testing*
- *Acceptance testing*
- *Performance testing*
- *Smoke testing*
- *Regression testing*
- *Production, staging and development systems*
- *Local development systems*
- *Remote development systems*
- *CI/CD pipelines*

056.2 Source Code Management (weight: 3)

Weight	2
---------------	---

Description	<i>The candidate should understand the main concepts of source code management. This includes the concept of source code repositories as well as branches, tags, forks and merges.</i>
--------------------	--

Key Knowledge Areas:

- *Understanding source code repositories (public and private)*
- *Understanding the principles of source code management and repository organization*
- *Awareness of common SCM systems (Git, Subversion, CVS)*
- *Awareness of the terms Version Control System (VCS), Revision Control System and Source Code Management systems (SCM)*

Partial list of the used files, terms, and utilities:

- *Source code repositories*
- *Commits, Branches and Tags*
- *Feature, development and release branches*
- *Subrepositories*
- *Code merges*

056.3 Communication and Collaboration Tools (weight: 2)

Weight	2
Description	<i>The candidate should understand the main categories and features of tools commonly used for communication and collaboration in software development projects.</i>

Key Knowledge Areas:

- *Understanding common tools for communication*
- *Understanding common ways to capture and secure knowledge*
- *Understanding common tools for information management and publication*
- *Understanding common types of documentation*

- *Understanding common collaboration features of source code management platforms*
- *Understanding the concepts of stand-alone, federated and centralized managed applications and platforms*

Partial list of the used files, terms, and utilities:

- *Instant messengers*
- *Chat platforms*
- *Mailing lists*
- *Newsletters*
- *Issue trackers and bug trackers*
- *Bug reports*
- *Merge requests and pull requests*
- *Helpdesk and ticketing systems*
- *Wikis*
- *Document Management Systems (DMS)*
- *Documentation websites*
- *Product websites*
- *Content Management Systems (CMS)*
- *Architecture documentation*
- *User documentation*
- *Administrator documentation*
- *Developer documentation*