



## Apprenticeship Program Occupational Standards

# Application Developer on Z Competency Framework

O\*NET CODE: 15-1132.00 (Software Developers, Application)  
Updated on 03/03/2020

This document was created by IBM's Apprenticeship Program as an open source standard to help industry accelerate their journey to developing new collar apprenticeship and work-based learning programs.

# Contents

## Table of Contents

<b>Work Process Schedule and Related Instruction Outline.....</b>	<b>3</b>
Overview .....	3
Work Processes .....	4
Competencies and Performance Criteria .....	5
Evidence Types .....	7
Foundational Performance Criteria.....	8
<b>Outline of Related Instruction .....</b>	<b>17</b>
Professional Foundational Instruction .....	17
Technical Foundational Instruction.....	17

# Work Process Schedule and Related Instruction Outline

## Overview

The occupational standards include the competency framework that outlines the required technical and professional competencies for each occupation. Demonstrated proficiency in all competencies is a requirement for successful completion of a competency based apprenticeship program.

The following section contains work process, training outline, and related instruction for the **IBM Application Developer on Z Apprenticeship**. The criteria are segregated into three distinct parts.

- **Part I: Work Process** – This section delineates the general outline of basic, high-level requirements that each participant will need to satisfy including projects, coaching, job shadowing, and training.
- **Part II: Competencies and Performance Criteria** – In support of this competency-based apprenticeship model, this section identifies what technical knowledge and professional behaviors will be evident as a product of achieving proficiency in these areas.
  - Competency Outcomes – These are the skills that apprentices are required to master during their apprenticeship. Competencies are defined as knowledge (K), behavior (B), or skills (S).
  - Assessment Criteria - Outlines the specific knowledge or combination of skills that each apprentice is required to learn to demonstrate proficiency.
  - Evidence types are the mechanisms used to evaluate apprentice's overall proficiency in a stated competency.
- **Part III: Outline of Related Instruction** – This section outlines specific formal training that each participant will be required to complete or demonstrate mastery.

## Work Processes

On the Job Training:	Validated by Manager / Mentor	Date
<b>Principles and Practices</b>		
1. Apply agile principles and practices		
2. Leverage IBM Design Thinking practices		
3. Apply business and professional acumen skills		
4. Use and contribute to shared and open GitHub repositories		
5. Gather and analyze data to draw insights		
<b>Software Engineering Fundamentals</b>		
6. Conduct software design and modeling		
7. Develop and write software code		
8. Perform software testing and problem solving		
9. Perform system scaling and security		
<b>DevOps Fundamentals</b>		
10. Perform continuous integration of code		
11. Perform continuous delivery of code		
12. Execute automation as part of the development lifecycle		
13. Implement metrics and measurement		
<b>Platforms, Services, and Solutions</b>		
14. Use various cloud tools, services, and platforms		
15. Use logging and monitoring tools		
16. Migrate data from on-premise solution to cloud solution		

## Competencies and Performance Criteria

### Foundational Competencies

#	Knowledge / Skill / Behavior	Description
<b>Principles and Practices</b>		
1.0	K / S / B	Understand, articulate, and demonstrate agile principles and practices
2.0	K / S / B	Understand, articulate, and demonstrate Design Thinking
3.0	B	Demonstrate key teamwork and collaborative behaviors
4.0	B	Demonstrate strong communication skills
5.0	K / B	Understand and model good feedback behaviors
6.0	K / B	Understand and demonstrate social coding behaviors
7.0	K / S	Demonstrate ability to analyze data sets, identify insights, and leverage to drive decision making
<b>Software Engineering Fundamentals</b>		
8.0	K	Demonstrate knowledge of key computer programming fundamentals
9.0	K / S	Understand and demonstrate key software design fundamentals
10.0	K / S	Understand and demonstrate test-driven development
11.0	K	Understand and navigate the complexity associated with enterprise-level development
12.0	K / S / B	Understand, articulate, and demonstrate clean coding behaviors
13.0	K / B	Understand and manage technical debt
14.0	K / S	Understand and demonstrate knowledge of web programming skills
15.0	K / S	Understand how to use version control for all elements of the software delivery lifecycle
16.0	K / S	Understand and demonstrate how to construct and test quality code, at scale
<b>DevOps Fundamentals</b>		
17.0	K / S	Understand and demonstrate continuous integration
18.0	K / S	Understand and demonstrate continuous delivery
19.0	K / S	Understand and demonstrate feature decoupling
20.0	K / S	Understand and demonstrate DevOps automation
21.0	K / S	Understand and demonstrate DevOps metrics and measurements
<b>Platforms, Services, and Solutions</b>		
22.0	K / S	Understand and demonstrate knowledge of cloud computing fundamentals, including the various tools, services, and principles
23.0	K / S	Understand and demonstrate the design patterns and practices for building cloud native services
24.0	K / S	Understand the relationship between scaling techniques, how to exploit them, and where

25.0	K	Understand the various platforms, their differences, relative strengths / weaknesses, and integrations
26.0	K	Understand the topologies of enterprise solutions, and how clients use our portfolio of products and services together
27.0	K / S	Understand and demonstrate the use of logging and monitoring tools and infrastructure
28.0	K	Understand the need for mobility and migration of data from on-premise to cloud solutions, and the implications
29.0	K / S	Understand and demonstrate knowledge of Enterprise computing platform
30.0	K / S	Demonstrate basic understanding of the hierarchical and relational DBMS
31.0	K / S	Understand the concepts, functions and facilities of CICS

## Evidence Types

Evidence Code	Description
O	Observation
Q&A	Questions and answers
P	Learner products
RA	Reflective accounts / personal statements
S	Simulation
PD	Professional discussion
A	Assignments, projects, case studies
MT	Mentor testimony
EW	Expert witness evidence
RPL	Recognition of prior learning

## Foundational Performance Criteria

Job Role	Application Developer on Z Specialist
O*NET Code	<a href="#">15.1132</a> (Software Developers, Application)
Apprenticeship Level	Foundation
Guided Learning Hours	290
Experiential Hours	1710

Competency Outcome	Assessment Criteria	Evidence Types	Education Material
1.0 Understand, articulate, and demonstrate agile principles and practices	1.1 Demonstrate knowledge of agile principles and how IBM implements them in a development environment 1.2 Lead and participate in agile planning activities, including story points, planning poker, Kanban boards, prioritizing work, and writing stories 1.3 Model “small batch” practices by successfully breaking down work tasks into smaller components 1.4 Lead and participate in retrospectives to drive continuous improvement	<input checked="" type="checkbox"/> O <input checked="" type="checkbox"/> Q&A <input type="checkbox"/> P <input type="checkbox"/> RA <input type="checkbox"/> S <input type="checkbox"/> PD <input checked="" type="checkbox"/> A <input checked="" type="checkbox"/> MT <input type="checkbox"/> EW <input checked="" type="checkbox"/> RPL	
2.0 Understand, articulate, demonstrate IBM Design Thinking	2.1 Demonstrate knowledge of IBM Design Thinking and how IBM implements it in a development environment 2.2 Create empathy maps for identified personas 2.3 Execute discovery phase to identify customer requirements 2.4 Develop hills that communicate project intent with clarity and flexibility 2.5 Conduct successful playbacks with stakeholders to exchange feedback and measure progress 2.6 Identify and leverage sponsor users to provide real-world perspective 2.7 Demonstrate behaviors and work plans aligned with fail fast / MVP principles	<input checked="" type="checkbox"/> O <input checked="" type="checkbox"/> Q&A <input checked="" type="checkbox"/> P <input checked="" type="checkbox"/> RA <input checked="" type="checkbox"/> S <input type="checkbox"/> PD <input checked="" type="checkbox"/> A <input checked="" type="checkbox"/> MT <input type="checkbox"/> EW <input checked="" type="checkbox"/> RPL	
3.0 Demonstrate teamwork and collaborative behaviors	3.1 Demonstrate ability to work co-operative with others 3.2 Demonstrate ability to work as part of a team 3.3 Demonstrate successful use of playbacks and reviews 3.4 Demonstrate goal-setting, being solution-focused, managing and strengthening relationships, and working with diverse perspectives.	<input checked="" type="checkbox"/> O <input type="checkbox"/> Q&A <input type="checkbox"/> P <input checked="" type="checkbox"/> RA <input checked="" type="checkbox"/> S <input checked="" type="checkbox"/> PD <input checked="" type="checkbox"/> A <input checked="" type="checkbox"/> MT <input checked="" type="checkbox"/> EW <input checked="" type="checkbox"/> RPL	



<p>4.0 Demonstrate strong communication skills</p>	<p>4.1 Demonstrate strong communication skills through the selection of the appropriate communication method for each message</p> <p>4.2 Demonstrate strong presentation skills through quality materials and clear presentation</p> <p>4.3 Demonstrate appropriate use of email for general communications</p> <p>4.4 Demonstrate appropriate use of Slack for general communications</p> <p>4.5 Demonstrate appropriate use of Slack as a collaboration tool</p>	<p><input checked="" type="checkbox"/> O</p> <p><input checked="" type="checkbox"/> Q&amp;A</p> <p><input type="checkbox"/> P</p> <p><input checked="" type="checkbox"/> RA</p> <p><input type="checkbox"/> S</p> <p><input checked="" type="checkbox"/> PD</p> <p><input checked="" type="checkbox"/> A</p> <p><input checked="" type="checkbox"/> MT</p> <p><input checked="" type="checkbox"/> EW</p> <p><input checked="" type="checkbox"/> RPL</p>	
<p>5.0 Understand and model good feedback behaviors</p>	<p>5.1 Understand the importance of feedback in all we do</p> <p>5.2 Successfully leverage Net Promoter Score methodology</p> <p>5.3 Deliver quality feedback to team members</p> <p>5.4 Receive feedback gracefully and act on it</p> <p>5.5 Model goal setting behaviors through performance management system</p> <p>5.6 Model a culture of feedback with all team members</p>	<p><input checked="" type="checkbox"/> O</p> <p><input checked="" type="checkbox"/> Q&amp;A</p> <p><input type="checkbox"/> P</p> <p><input type="checkbox"/> RA</p> <p><input checked="" type="checkbox"/> S</p> <p><input checked="" type="checkbox"/> PD</p> <p><input checked="" type="checkbox"/> A</p> <p><input checked="" type="checkbox"/> MT</p> <p><input checked="" type="checkbox"/> EW</p> <p><input checked="" type="checkbox"/> RPL</p>	
<p>6.0 Understand and demonstrate social coding behaviors</p>	<p>6.1 Demonstrate knowledge of social coding behaviors</p> <p>6.2 Execute successful "Fork and Pull" model in GitHub</p> <p>6.3 Demonstrate effective searching via GitHub / GHE to identify code</p> <p>6.4 Leverage established channels, i.e. slack, to collaborate and identify code</p> <p>6.5 Explain the use of Open Source and the implications</p> <p>6.6 Demonstrate successful usage of Open Sources scanning tools</p> <p>6.7 Articulate the appropriate protection and usage of Intellectual Property</p>	<p><input checked="" type="checkbox"/> O</p> <p><input checked="" type="checkbox"/> Q&amp;A</p> <p><input type="checkbox"/> P</p> <p><input type="checkbox"/> RA</p> <p><input checked="" type="checkbox"/> S</p> <p><input checked="" type="checkbox"/> PD</p> <p><input checked="" type="checkbox"/> A</p> <p><input checked="" type="checkbox"/> MT</p> <p><input checked="" type="checkbox"/> EW</p> <p><input checked="" type="checkbox"/> RPL</p>	<p><a href="#">Writing Clean Code Humans</a></p> <p><a href="#">GitHub Fundamentals</a></p>
<p>7.0 Demonstrate ability to analyze data sets, identify insights, and leverage to drive decision making</p>	<p>7.1 Understand the value of real-time data gathering to reduce risk and increase success</p> <p>7.2 Articulate the different types of analytics: predictive, prescriptive, descriptive, cognitive, machine learning</p> <p>7.3 Demonstrate use of data analytics to make decisions</p> <p>7.4 Successfully analyze large data sets and leverage data to drive insights</p>	<p><input checked="" type="checkbox"/> O</p> <p><input checked="" type="checkbox"/> Q&amp;A</p> <p><input type="checkbox"/> P</p> <p><input type="checkbox"/> RA</p> <p><input type="checkbox"/> S</p> <p><input checked="" type="checkbox"/> PD</p> <p><input checked="" type="checkbox"/> A</p> <p><input checked="" type="checkbox"/> MT</p> <p><input checked="" type="checkbox"/> EW</p> <p><input type="checkbox"/> RPL</p>	<p><a href="#">Data Analytics hands-on</a></p> <p><a href="#">Big Data 101</a></p> <p><a href="#">Hadoop 101</a></p> <p><a href="#">Spark Fundamentals</a></p>
<p>8.0 Demonstrate knowledge of key computer programming fundamentals</p>	<p>8.1 Demonstrate an understanding of the history of computing</p> <p>8.2 Demonstrate an understanding of the history and principles of software engineering</p>	<p><input checked="" type="checkbox"/> O</p> <p><input checked="" type="checkbox"/> Q&amp;A</p> <p><input type="checkbox"/> P</p> <p><input checked="" type="checkbox"/> RA</p> <p><input type="checkbox"/> S</p>	<p><a href="#">TCP/IP Networking for Developers</a></p>

	<p>8.3 Demonstrate an understanding of the history and principles of data management</p> <p>8.4 Demonstrate an understanding of the history and principles of networking fundamentals</p> <p>8.5 Demonstrate an understanding of the history and principles of infrastructure</p> <p>8.6 Demonstrate an understanding of the history and principles of integrated development environments</p>	<input type="checkbox"/> PD <input type="checkbox"/> A <input checked="" type="checkbox"/> MT <input type="checkbox"/> EW <input checked="" type="checkbox"/> RPL	<a href="#">AP Computer Science Principles</a>
<p>9.0 Understand and demonstrate key software design fundamentals</p>	<p>9.1 Demonstrate knowledge and understanding of key software design fundamentals (for e.g. SDLC)</p> <p>9.2 Demonstrate successful problem solving behaviors</p> <p>9.2.1 Demonstrate ability to decompose a problem</p> <p>9.2.2 Demonstrate ability to use abstract data types</p> <p>9.2.3 Demonstrate ability to apply algorithms as a solution</p> <p>9.2.4 Demonstrate ability to follow logical conclusions</p> <p>9.3 Demonstrate the use of an object-oriented language or modelling language to design a model that promotes clean code</p> <p>9.3.1 Successfully decompose into discrete components in a coherent design that will enable the software to grow and be maintained efficiently</p> <p>9.4 Demonstrate the use of functional programming skills</p> <p>9.5 Demonstrate ability to recognize and apply design patterns to address a problem</p> <p>Demonstrate proficiency in languages like COBOL, PL/1, Assembler, Java, Python, or JavaScript</p>	<input checked="" type="checkbox"/> O <input checked="" type="checkbox"/> Q&A <input checked="" type="checkbox"/> P <input type="checkbox"/> RA <input checked="" type="checkbox"/> S <input type="checkbox"/> PD <input checked="" type="checkbox"/> A <input checked="" type="checkbox"/> MT <input type="checkbox"/> EW <input checked="" type="checkbox"/> RPL	<a href="#">SW Design Fundamentals</a>  <a href="#">Java Fundamentals</a>  <a href="#">Python Fundamentals</a>  <a href="#">COBOL Programming with VSCode</a>  <a href="#">Javascript Fundamentals</a>  <a href="#">Node.js – Getting Started</a>  <a href="#">Software Architecture Fundamentals</a>
<p>10.0 Understand and demonstrate test-driven development</p>	<p>10.1 Articulate the differences and relative strengths / weakness of the various models of test-driven or behavior driven development</p> <p>10.2 Demonstrate use of various models of test-driven or behavior driven development</p> <p>10.3 Demonstrate understanding of testing life cycle and levels of testing (for e.g: Unit test, Function test (Green screen testing), System Test, Integration test, Regression test, etc)</p>	<input checked="" type="checkbox"/> O <input checked="" type="checkbox"/> Q&A <input checked="" type="checkbox"/> P <input type="checkbox"/> RA <input checked="" type="checkbox"/> S <input type="checkbox"/> PD <input checked="" type="checkbox"/> A <input checked="" type="checkbox"/> MT <input type="checkbox"/> EW <input checked="" type="checkbox"/> RPL	<a href="#">Test Driven Development – Big Picture</a>
<p>11.0 Understand and navigate the complexity associated with enterprise-level development</p>	<p>11.1 Articulate the complexities and challenges associated with enterprise-scale development</p> <p>11.2 Understand how to apply security at enterprise</p> <p>11.3 Understand how to scale solutions</p> <p>11.4 Understand how to manage performance at enterprise</p> <p>11.5 Understand how to address legacy integrations</p>	<input checked="" type="checkbox"/> O <input checked="" type="checkbox"/> Q&A <input checked="" type="checkbox"/> P <input type="checkbox"/> RA <input checked="" type="checkbox"/> S <input type="checkbox"/> PD <input type="checkbox"/> A	<a href="#">Clean Architecture – Patterns, Practices &amp; Principles</a>

		<input checked="" type="checkbox"/> MT <input type="checkbox"/> EW <input checked="" type="checkbox"/> RPL	<a href="#">Understanding Enterprise Architecture</a>  <a href="#">TOGAF 9.1 Overview</a>
12.0 Understand, articulate, and demonstrate clean coding behaviors	12.1 Demonstrate use of refactoring to yield clean code 12.2 Demonstrate ability to write code with clarity 12.3 Demonstrate ability to write code that minimizes duplication 12.4 Demonstrate ability to write code with simplicity 12.5 Explain the various clean coding principles represented by the acronym SOLID 12.6 Demonstrate use of SOLID principles in code	<input checked="" type="checkbox"/> O <input checked="" type="checkbox"/> Q&A <input checked="" type="checkbox"/> P <input type="checkbox"/> RA <input checked="" type="checkbox"/> S <input type="checkbox"/> PD <input type="checkbox"/> A <input checked="" type="checkbox"/> MT <input type="checkbox"/> EW <input checked="" type="checkbox"/> RPL	<a href="#">Writing Clean Code Humans</a>
13.0 Understand and manage technical debt	13.1 Describe the issues caused by technical debt 13.2 Demonstrate ability to identify technical debt 13.3 Demonstrate ability to eliminate technical debt 13.4 Articulate and/or demonstrate ability to use code scanning and static analysis tools	<input checked="" type="checkbox"/> O <input checked="" type="checkbox"/> Q&A <input checked="" type="checkbox"/> P <input type="checkbox"/> RA <input type="checkbox"/> S <input type="checkbox"/> PD <input type="checkbox"/> A <input checked="" type="checkbox"/> MT <input type="checkbox"/> EW <input checked="" type="checkbox"/> RPL	<a href="#">Writing Clean Code Humans</a>  <a href="#">Understanding and Eliminating Technical Debt</a>
14.0 Understand and demonstrate knowledge of web programming skills	14.1 Understand and articulate the value of distributed, modern computing architecture and the role it plays in modern service development 14.2 Demonstrate proficient back-end or front end development skills 14.3 Demonstrate basic knowledge of back-end and front end development 14.4 Demonstrate understanding of how to scale applications using database 14.5 Demonstrate understanding of how to build and deploy scalable applications	<input checked="" type="checkbox"/> O <input checked="" type="checkbox"/> Q&A <input checked="" type="checkbox"/> P <input type="checkbox"/> RA <input type="checkbox"/> S <input type="checkbox"/> PD <input checked="" type="checkbox"/> A <input checked="" type="checkbox"/> MT <input type="checkbox"/> EW <input checked="" type="checkbox"/> RPL	<a href="#">Front end development Quick Start</a>  <a href="#">Introduction to web development</a>  <a href="#">jQuery Fundamentals</a>  <a href="#">AngularJS Fundamentals</a>  <a href="#">Designing RESTful web APIs</a>  <a href="#">Practical Start to React</a>

			<a href="#">Building applications with React and Flux</a>
15.0 Understand how to use version control for all elements of the software delivery lifecycle	15.1 Describe the principles of version control 15.2 Describe versioning, releases, issues, and merges in source code 15.3 Demonstrate proper usage of versioning, releases, issues, and merges in source code 15.4 Understand the use of dependencies and libraries from within and outside the team 15.5 Demonstrate usage of dependencies and libraries from within and outside the team	<input checked="" type="checkbox"/> O <input checked="" type="checkbox"/> Q&A <input checked="" type="checkbox"/> P <input type="checkbox"/> RA <input type="checkbox"/> S <input type="checkbox"/> PD <input checked="" type="checkbox"/> A <input checked="" type="checkbox"/> MT <input type="checkbox"/> EW <input checked="" type="checkbox"/> RPL	OJT
16.0 Understand and demonstrate how to construct and test quality code, at scale	16.1 Understand how to construct quality code at scale 16.2 Understand the techniques used for ensuring quality 16.3 Understand the different types of testing, where to apply them, and the relative amounts 16.4 Understand the successful use of test coverage tools 16.5 Understand successful use of performance testing 16.6 Understand successful use of boundary and limit testing Understand successful use of chaotic testing	<input checked="" type="checkbox"/> O <input checked="" type="checkbox"/> Q&A <input checked="" type="checkbox"/> P <input type="checkbox"/> RA <input checked="" type="checkbox"/> S <input type="checkbox"/> PD <input checked="" type="checkbox"/> A <input checked="" type="checkbox"/> MT <input type="checkbox"/> EW <input checked="" type="checkbox"/> RPL	OJT
17.0 Understand and demonstrate continuous integration	17.1 Articulate the value of continuous integration in a development environment 17.2 Demonstrate CI principles through frequent commits to trunk of codebase 17.3 Demonstrate CI principles through use of automated tests to ensure codebase is in a good state 17.4 Demonstrate proficient use of continuous integration tools 17.5 Demonstrate understanding of the impact of commit defects into shared codebase	<input checked="" type="checkbox"/> O <input checked="" type="checkbox"/> Q&A <input checked="" type="checkbox"/> P <input type="checkbox"/> RA <input type="checkbox"/> S <input type="checkbox"/> PD <input type="checkbox"/> A <input checked="" type="checkbox"/> MT <input type="checkbox"/> EW <input checked="" type="checkbox"/> RPL	<a href="#">Continuous Integration and Delivery – Big Picture</a>  <a href="#">Getting Started with Jenkins 2</a>  <a href="#">Travis CI</a>
18.0 Understand and demonstrate continuous delivery	18.1 Articulate the value of continuous delivery in a development environment 18.2 Understand CD principles through short, frequent, automated delivery of code 18.3 Demonstrate proficient use of continuous delivery tools	<input checked="" type="checkbox"/> O <input checked="" type="checkbox"/> Q&A <input checked="" type="checkbox"/> P <input type="checkbox"/> RA <input type="checkbox"/> S <input type="checkbox"/> PD <input type="checkbox"/> A <input checked="" type="checkbox"/> MT <input type="checkbox"/> EW <input checked="" type="checkbox"/> RPL	<a href="#">Continuous Integration and Delivery – Big Picture</a>  <a href="#">Getting Started with Jenkins 2</a>  <a href="#">Travis CI</a>
19.0 Understand and demonstrate feature decoupling	19.1 Articulate the value of feature decoupling in a development environment 19.2 Demonstrate ability to proficiently use feature toggles	<input checked="" type="checkbox"/> O <input checked="" type="checkbox"/> Q&A	<a href="#">Feature Toggles</a>

	19.3 Understand knowledge of microservice architecture	<input checked="" type="checkbox"/> P <input type="checkbox"/> RA <input type="checkbox"/> S <input type="checkbox"/> PD <input type="checkbox"/> A <input checked="" type="checkbox"/> MT <input type="checkbox"/> EW <input checked="" type="checkbox"/> RPL	
20.0 Understand and demonstrate DevOps automation	20.1 Articulate the value of automation to the development lifecycle 20.2 Demonstrate knowledge of the tools required to execute automation in a DevOps development team 20.3 Demonstrate use of automated run-books to minimize call-out for service failures 20.4 Demonstrate proficient use of UCD 20.5 Demonstrate proficient use of Ansible and scripted automation 20.6 Demonstrate proficient use of Chef	<input checked="" type="checkbox"/> O <input checked="" type="checkbox"/> Q&A <input checked="" type="checkbox"/> P <input type="checkbox"/> RA <input type="checkbox"/> S <input type="checkbox"/> PD <input checked="" type="checkbox"/> A <input type="checkbox"/> MT <input type="checkbox"/> EW <input checked="" type="checkbox"/> RPL	<a href="#">DevOps Big Picture</a>  <a href="#">Implementing DevOps in the real world</a>
21.0 Understand and demonstrate DevOps metrics and measurements	21.1 Understand the use and articulate the value of instrumentation and automated measurements to enable continuous improvement 21.2 Understand the various methodologies used to measure and track progress	<input checked="" type="checkbox"/> O <input checked="" type="checkbox"/> Q&A <input checked="" type="checkbox"/> P <input type="checkbox"/> RA <input type="checkbox"/> S <input type="checkbox"/> PD <input type="checkbox"/> A <input checked="" type="checkbox"/> MT <input type="checkbox"/> EW <input checked="" type="checkbox"/> RPL	<a href="#">DevOps Big Picture</a>  <a href="#">Implementing DevOps in the real world</a>
22.0 Understand and demonstrate knowledge of cloud computing fundamentals, including the various tools, services, and principles	22.1 Articulate the value of cloud infrastructures as it relates to efficiency, scalability, and resiliency of services 22.2 Articulate the value and use of containers in cloud computing 22.3 Demonstrate proficient use of containers like Docker and Vagrant 22.4 Articulate the value of platforms like OpenStack and IaaS 22.5 Demonstrate proficient use of cloud platforms 22.6 Articulate the value of container management systems 22.7 Demonstrate proficient use of service development practices, including adherence to 12-factor app rules	<input checked="" type="checkbox"/> O <input checked="" type="checkbox"/> Q&A <input checked="" type="checkbox"/> P <input type="checkbox"/> RA <input type="checkbox"/> S <input type="checkbox"/> PD <input checked="" type="checkbox"/> A <input checked="" type="checkbox"/> MT <input type="checkbox"/> EW <input checked="" type="checkbox"/> RPL	<a href="#">IBM Cloud Essentials</a>  <a href="#">IBM Cloud Application Development V3</a>  <a href="#">IBM Cloudant</a>  <a href="#">Docker Containers – Big Picture</a>  <a href="#">Container &amp; Kubernetes Essentials</a>

			<a href="#">Introduction to OpenStack</a>
23.0 Understand and demonstrate the design patterns and practices for building cloud native services	<p>23.1 Articulate the design patterns and practices for building highly scalable, resilient services and their lifecycles</p> <p>23.2 Demonstrate understanding of Kubernetes</p> <p>23.3 Demonstrate understanding of microservices architecture, including platforms like Liber8 and ISTIO</p> <p>23.4 Understand proficient understanding of circuit-breaker patterns like Hystrix and ISTIO</p>	<input checked="" type="checkbox"/> O <input checked="" type="checkbox"/> Q&A <input checked="" type="checkbox"/> P <input type="checkbox"/> RA <input type="checkbox"/> S <input type="checkbox"/> PD <input checked="" type="checkbox"/> A <input checked="" type="checkbox"/> MT <input type="checkbox"/> EW <input checked="" type="checkbox"/> RPL	<a href="#">Get Started with Microservices ISTIO</a>  <a href="#">Beyond Basics ISTIO</a>
24.0 Understand the relationship between scaling techniques, how to exploit them, and where	<p>24.1 Articulate the relationship between scaling techniques, how to exploit them, and where</p> <p>24.2 Articulate the value of scale-out vs. scale-up</p> <p>24.3 Demonstrate use of scale-out vs. scale-up</p> <p>24.4 Understand the value of fail fast / recovery</p> <p>24.5 Understand the use of fail fast / recovery</p>	<input checked="" type="checkbox"/> O <input checked="" type="checkbox"/> Q&A <input type="checkbox"/> P <input type="checkbox"/> RA <input type="checkbox"/> S <input type="checkbox"/> PD <input type="checkbox"/> A <input checked="" type="checkbox"/> MT <input type="checkbox"/> EW <input checked="" type="checkbox"/> RPL	
25.0 Understand the various platforms, their differences, relative strengths / weaknesses, and integrations	<p>25.1 Outline the various platforms, their differences, relative strengths / weaknesses, and integrations</p> <p>25.2 Demonstrate basic knowledge of zOS</p> <p>25.3 Demonstrate basic knowledge of Linux</p> <p>25.4 Demonstrate basic knowledge of Mobile</p> <p>25.5 Demonstrate basic knowledge of IoT</p>	<input checked="" type="checkbox"/> O <input checked="" type="checkbox"/> Q&A <input checked="" type="checkbox"/> P <input type="checkbox"/> RA <input type="checkbox"/> S <input type="checkbox"/> PD <input checked="" type="checkbox"/> A <input checked="" type="checkbox"/> MT <input type="checkbox"/> EW <input checked="" type="checkbox"/> RPL	<a href="#">z/OS Introduction - Redbook Video Course</a>
26.0 Understand the topologies of enterprise solutions, and how clients use our portfolio of products and services together	<p>26.1 Articulate the topologies of enterprise solutions, and how clients use our portfolio of products and services together</p> <p>26.2 Articulate the common use cases for APIs</p> <p>26.3 Successfully design, publish, and manage APIs</p> <p>26.4 Demonstrate understanding of zOS Connect &amp; use of APIs to expose data</p>	<input checked="" type="checkbox"/> O <input checked="" type="checkbox"/> Q&A <input checked="" type="checkbox"/> P <input type="checkbox"/> RA <input type="checkbox"/> S <input type="checkbox"/> PD <input type="checkbox"/> A <input checked="" type="checkbox"/> MT <input type="checkbox"/> EW <input checked="" type="checkbox"/> RPL	<a href="#">Building an API Bridge</a>  <a href="#">Introduction to z/OS Connect</a>
27.0 Understand and demonstrate the use of logging and monitoring tools and infrastructure	<p>27.1 Articulate the value of logging and monitoring in an operational environment</p> <p>27.2 Demonstrate proficient use of logging and monitoring tools and infrastructure</p>	<input checked="" type="checkbox"/> O <input checked="" type="checkbox"/> Q&A <input checked="" type="checkbox"/> P <input type="checkbox"/> RA <input type="checkbox"/> S	<a href="#">z/OS Introduction - Redbook Video Course</a>

	<p>27.3 Demonstrate ability to implement and/or comply with logging policies</p> <p>27.4 Demonstrate ability to use dashboards for monitoring</p> <p>27.5 Demonstrate understanding of how data traverses</p>	<input type="checkbox"/> PD <input type="checkbox"/> A <input checked="" type="checkbox"/> MT <input type="checkbox"/> EW <input checked="" type="checkbox"/> RPL	
28.0 Understand the need for mobility and migration of data from on-premise to cloud solutions, and the implications	<p>28.1 Articulate the value in migrating data from on-premise to cloud</p> <p>28.2 Articulate the challenges in migrating data from on-premise to cloud</p> <p>28.3 Outline the process required to migrate data</p> <p>28.4 Demonstrate understanding of how to work with system of record data on Z and expose it</p>	<input checked="" type="checkbox"/> O <input checked="" type="checkbox"/> Q&A <input checked="" type="checkbox"/> P <input type="checkbox"/> RA <input type="checkbox"/> S <input type="checkbox"/> PD <input type="checkbox"/> A <input checked="" type="checkbox"/> MT <input type="checkbox"/> EW <input checked="" type="checkbox"/> RPL	OJT
29.0 Demonstrate basic understanding of Mainframe terminology and basic concepts	<p>29.1 Demonstrate ability to describe and compare various IBM Z components:</p> <p>29.1.1 Frame layout and cage usage</p> <p>29.1.2 Server models, books, memory, and cache structure</p> <p>29.1.3 Performance and millions of service units (MSUs)</p>	<input checked="" type="checkbox"/> O <input checked="" type="checkbox"/> Q&A <input type="checkbox"/> P <input checked="" type="checkbox"/> RA <input type="checkbox"/> S <input checked="" type="checkbox"/> PD <input checked="" type="checkbox"/> A <input checked="" type="checkbox"/> MT <input checked="" type="checkbox"/> EW <input type="checkbox"/> RPL	<a href="#">IBM z/OS Mainframe Practitioner</a>
30.0 Demonstrate basic understanding of the hierarchical and relational DBMS	<p>30.1 Demonstrate ability to articulate how databases are used in a typical online business</p> <p>30.2 Demonstrate ability to describe models for network connectivity for large systems</p> <p>30.3 Demonstrate ability to explain the role of DB2 in online transaction processing</p> <p>30.4 Demonstrate ability to list common DB2 data structures</p> <p>30.5 Demonstrate ability to describe how SQL works on z/OS</p> <p>30.6 Demonstrate ability to give an overview of application programming with DB2</p> <p>30.7 Demonstrate ability to explain what the IMS components are</p> <p>30.8 Demonstrate ability to describe the structure of the IMS DB subsystem</p>	<input checked="" type="checkbox"/> O <input checked="" type="checkbox"/> Q&A <input type="checkbox"/> P <input checked="" type="checkbox"/> RA <input type="checkbox"/> S <input checked="" type="checkbox"/> PD <input checked="" type="checkbox"/> A <input checked="" type="checkbox"/> MT <input checked="" type="checkbox"/> EW <input type="checkbox"/> RPL	<a href="#">Db2 Fundamentals V11</a>  <a href="#">Managing Db2 Operations V11</a>  <a href="#">IMS 15 Introduction</a>
31.0 Demonstrate basic understanding of the concepts, functions and facilities of CICS	<p>31.1 Demonstrate ability to describe the role of large systems in a typical online business</p> <p>31.2 Demonstrate ability to list the attributes common to most transaction systems</p> <p>31.3 Demonstrate ability to explain the role of CICS in online transaction processing</p> <p>31.4 Demonstrate ability to describe CICS programs, CICS transactions, and CICS tasks</p>	<input checked="" type="checkbox"/> O <input checked="" type="checkbox"/> Q&A <input type="checkbox"/> P <input checked="" type="checkbox"/> RA <input type="checkbox"/> S <input type="checkbox"/> PD <input checked="" type="checkbox"/> A <input checked="" type="checkbox"/> MT <input checked="" type="checkbox"/> EW	<a href="#">IBM CICS Video Course Series</a>  <a href="#">CICS Introduction</a>  <a href="#">CICS Explorer Fundamentals</a>

	31.5 Demonstrate ability to describe the CICS IMS components	<input type="checkbox"/> RPL	<a href="#">CICS Command Simulation 5.3</a>
--	--------------------------------------------------------------	------------------------------	---------------------------------------------



## Outline of Related Instruction

Professional Foundational Instruction is required for all apprentices.

### Professional Foundational Instruction

Professional Foundational Instruction	Formal Training Hours
Employee Onboarding and Integration Success Skills	50

### Technical Foundational Instruction

Technical Foundational Instruction	Formal Training Hours
<b>Principal and Practices</b> <ul style="list-style-type: none"> <li>• Agile practices, including operations, program fundamentals, and project and change management</li> <li>• Leadership skills</li> <li>• Big Data and Analytics</li> </ul>	24
<b>Software Engineering Fundamentals</b> <ul style="list-style-type: none"> <li>• Web development fundamentals</li> <li>• Clean coding and social coding behaviors and practices</li> <li>• Introductions to functional and imperative languages</li> <li>• COBOL programming</li> <li>• Test Driven Development</li> </ul>	92
<b>DevOps Fundamentals</b> <ul style="list-style-type: none"> <li>• Continuous integration and continuous delivery practices</li> <li>• Feature Decoupling</li> <li>• DevOps tools, including Chef, Travis, Ansible, and Jenkins</li> </ul>	11
<b>Platforms, Services, and Solutions</b> <ul style="list-style-type: none"> <li>• Cloud computing fundamentals</li> <li>• Enterprise solutions</li> </ul>	113
<b>Total</b>	<b>240</b>