

# Latest Study Workday-Pro-Integrations Questions, Workday-Pro-Integrations Exam Outline

---

## Workday HCM Pro Exam - Business Process Fundamentals Knowledge Questions and Answers Guide (already passed) Latest 2025

How can you easily find a business process definition in the tenant?

- A. task: <name of definition>
- B. process: <name of definition>
- C. bp: <name of definition>
- D. run: <name of definition> - **Correct Answers** C. bp: <name of definition>

What step types can you configure on a business process definition? (Select 2 **Correct Answers** .)

- A. Checklist step
- B. Assign step
- C. Integration step
- D. Security step - **Correct Answers** A. Checklist step
- C. Integration step

Where do you configure the business process?

- A. Business process policy
- B. Business process step
- C. Business process definition
- D. Tenant Setup - Business Process - **Correct Answers** C. Business process definition

P.S. Free 2025 Workday Workday-Pro-Integrations dumps are available on Google Drive shared by ITExamDownload:  
[https://drive.google.com/open?id=1pF9Km-Q1cywqdLnS2e9ulk\\_e74Ipr\\_C](https://drive.google.com/open?id=1pF9Km-Q1cywqdLnS2e9ulk_e74Ipr_C)

Before clients buy our Workday-Pro-Integrations questions torrent they can download them and try out them freely. The pages of our product provide the demo and the aim is to let the client know part of our titles before their purchase and what form our Workday-Pro-Integrations guide torrent is. The pages introduce the quantity of our questions and answers of our Workday-Pro-Integrations Guide Torrent. After you try out the free demo you could decide whether our Workday-Pro-Integrations exam torrent is worthy to buy or not. So you needn't worry that you will waste your money or our Workday-Pro-Integrations exam torrent is useless and boosts no values.

We take so much pride in the high pass rate of our Workday-Pro-Integrations study questions because according to the statistics from the feedbacks of all of our customers, under the guidance of our Workday-Pro-Integrations exam materials the pass rate has reached as high as 98% to 100%, which marks the highest pass rate in the field. So if you really want to pass the Workday-Pro-Integrations Exam as well as getting the certification with no danger of anything going wrong, just feel rest assured to buy our Workday-Pro-Integrations learning guide.

>> Latest Study Workday-Pro-Integrations Questions <<

**Workday-Pro-Integrations – 100% Free Latest Study Questions | the Best**

## Workday Pro Integrations Certification Exam Exam Outline

It is never too late to try new things no matter how old you are. Someone always give up their dream because of their ages, someone give up trying to overcome Workday-Pro-Integrations exam because it was difficult for them. Now, no matter what the reason you didn't pass the exam, our study materials will try our best to help you. If you are not sure what kinds of Workday-Pro-Integrations Exam Question is appropriate for you, you can try our free demo of the PDF version. There must be one that suits you best. Your life will become more meaningful because of your new change, and our Workday-Pro-Integrations question torrents will be your first step.

## Workday Pro Integrations Certification Exam Sample Questions (Q23-Q28):

### NEW QUESTION # 23

This is the XML file generated from a Core Connector; Positions integration.

```
1. <ps:Positions xmlns:ps="urn:com.workday/coreconnector/positions">
2.   <ps:Header>
3.     <ps:Prior_Entry_Time xsi:nil="true"/>
4.     <ps:Current_Entry_Time>2024-05-21T09:50:05.421-07:00</ps:Current_Entry_Time>
5.     <ps:Prior_Effective_Time xsi:nil="true"/>
6.     <ps:Current_Effective_Time>2024-05-21T00:00:00.000-07:00</ps:Current_Effective_Time>
7.     <ps:Full_File>true</ps:Full_File>
8.     <ps:Document_Retention_Policy>30</ps:Document_Retention_Policy>
9.     <ps:Position_Count>11</ps:Position_Count>
10.  </ps:Header>
11.  <ps:Position>
12.    <ps:Position_Data>
13.      <ps:Position_ID>P-00030</ps:Position_ID>
14.      <ps:Supervisory_Organization>SUPERVISORY_IT_HelpDesk_Department</ps:Supervisory_Organization>
15.      <ps:Job_Posting_Title>Senior IT Analyst</ps:Job_Posting_Title>
16.      <ps:Available_For_Hire>true</ps:Available_For_Hire>
17.      <ps:Availability_Date>2022-01-03</ps:Availability_Date>
18.      <ps:Location>San Francisco</ps:Location>
19.      <ps:Worker_Type>EE</ps:Worker_Type>
20.    </ps:Position_Data>
21.    <ps:Additional_Information>
22.      <ps:Reference_ID>P-00030</ps:Reference_ID>
23.      <ps:WID>73b5d48562a049b1820f5518469790b5</ps:WID>
24.    </ps:Additional_Information>
25.  </ps:Position>
26. </ps:Positions>
```

When performing an XSLT Transformation on the Core Connector: Positions XML output file, you want to show a hyperlink of positions that are not available for hiring as an entry in the Message tab.

What are all the needed ETV items to meet the above requirements?

- A. 

```
<etv:minLength="0"
etv:target="{ps:Additional_Information/ps:WID}"
```
- B. 

```
<etv:required="true"
etv:targetWID="{ps:Additional_Information/ps:WID}"
```
- C. 

```
<etv:minLength="0"
etv:targetWID="{ps:Additional_Information/ps:WID}"
```
- D. 

```
<etv:required="true"
etv:target="{ps:Additional_Information/ps:WID}"
```

**Answer: D**

**Explanation:**

In Workday integrations, the Extension for Transformation and Validation (ETV) framework is used within XSLT transformations to apply validation and formatting rules to XML data, such as the output from a Core Connector (e.g., Positions integration). In this scenario, you need to perform an XSLT transformation on the Core Connector: Positions XML output file to display a hyperlink for

positions that are not available for hiring as an entry in the Message tab. This requires configuring ETV attributes to ensure the data is present and correctly targeted for the hyperlink.

Here's why option B is correct:

\* Requirement Analysis: The requirement specifies showing a hyperlink for positions "not available for hiring." In the provided XML, the `ps:Available_For_Hire` field under `ps:Position_Data` indicates whether a position is available for hire (e.g., `<ps:Available_For_Hire>true</ps:Available_For_Hire>`).

For positions where this is false, you need to create a message (hyperlink) in the Message tab, which typically requires linking to a Workday ID (WID) or other identifier.

\* ETV Attributes:

\* `etv:required="true"`: This ensures that the `ps:WID` value under `ps:Additional_Information` is mandatory for the transformation. If the WID is missing, the transformation will fail or generate an error, ensuring that the hyperlink can be created only for valid positions with an associated WID.

\* `etv:target="[ps:Additional_Information/ps:WID]"`: This specifies that the target of the transformation (e.g., the hyperlink) should be the WID value found at `ps:Additional_Information/ps:WID` in the XML. This WID can be used to construct a hyperlink to the position in Workday, meeting the requirement to show a hyperlink for positions not available for hiring.

\* Context in XML: The XML shows `ps:Additional_Information` containing `ps:WID` (e.g., `<ps:WID>73bd4d8562e04b1820f55818467905b</ps:WID>`), which is a unique identifier for the position.

By targeting this WID with `etv:target`, you ensure the hyperlink points to the correct position record in Workday when `ps:Available_For_Hire` is false.

Why not the other options?

\* A.

\* `etv:minLength="0"`

\* `etv:targetWID="[ps:Additional_Information/ps:WID]"`

\* `etv:minLength="0"` allows the WID to be empty or have zero length, which contradicts the need for a valid WID to create a hyperlink. It does not ensure the data is present, making it unsuitable.

Additionally, `etv:targetWID` is not a standard ETV attribute; the correct attribute is `etv:target`, making this option incorrect.

\* C.

\* `etv:minLength="0"`

\* `etv:target="[ps:Additional_Information/ps:WID]"`

\* Similar to option A, `etv:minLength="0"` allows the WID to be empty, which does not meet the requirement for a mandatory WID to create a hyperlink. This makes it incorrect, as the hyperlink would fail if the WID is missing.

\* D.

\* `etv:required="true"`

\* `etv:targetWID="[ps:Additional_Information/ps:WID]"`

\* While `etv:required="true"` ensures the WID is present, `etv:targetWID` is not a standard ETV attribute. The correct attribute is `etv:target`, making this option syntactically incorrect and unsuitable for the transformation.

To implement this in XSLT for a Workday integration:

\* Use the ETV attributes from option B (`etv:required="true"` and `etv:target="[ps:Additional_Information/ps:WID]"`) within your XSLT template to validate and target the `ps:WID` for positions where `ps:Available_For_Hire` is false.

This ensures the transformation generates a valid hyperlink in the Message tab, linking to the position's WID in Workday.

References:

\* Workday Pro Integrations Study Guide: Section on "ETV in XSLT Transformations" - Details the use of ETV attributes like `required` and `target` for validating and targeting data in Workday XML, including handling identifiers like WID for hyperlinks.

\* Workday Core Connector and EIB Guide: Chapter on "XML Transformations" - Explains how to use ETV attributes in XSLT to process position data, including creating messages or hyperlinks based on conditions like `Available_For_Hire`.

\* Workday Integration System Fundamentals: Section on "ETV for Message Generation" - Covers applying ETV attributes to generate hyperlinks in the Message tab, ensuring data integrity and correct targeting of Workday identifiers like WID.

## NEW QUESTION # 24

What is the purpose of the `<xsl:template>` element?

- A. Generate an output file name.
- **B. Provide rules to apply to a specified node.**
- C. Determine the output file type.
- D. Grant access to the XSLT language.

**Answer: B**

Explanation:

The `<xsl:template>` element is a fundamental component of XSLT (Extensible Stylesheet Language Transformations), which is widely used in Workday integrations, particularly within document transformation systems such as those configured via the Enterprise Interface Builder (EIB) or Document Transformation Connectors. Its primary purpose is to define rules or instructions that dictate how specific nodes in an XML source document should be processed and transformed into the desired output format.

Here's a detailed explanation of why this is the correct answer:

- \* In XSLT, the `<xsl:template>` element is used to create reusable transformation rules. It typically includes a `match` attribute, which specifies the XML node or pattern (e.g., an element, attribute, or root node) to which the template applies. For example, `<xsl:template match="Employee">` would target all `<Employee>` elements in the source XML.

- \* Inside the `<xsl:template>` element, you define the logic—such as extracting data, restructuring it, or applying conditions—that determines how the matched node is transformed into the output. This makes it a core mechanism for controlling the transformation process in Workday integrations.

- \* In the context of Workday, where XSLT is often used to reformat XML data into formats like CSV, JSON, or custom XML for external systems, `<xsl:template>` provides the structure for specifying how data from Workday's XML output (e.g., payroll or HR data) is mapped and transformed.

Let's evaluate why the other options are incorrect:

- \* A. Determine the output file type: The `<xsl:template>` element does not control the output file type (e.g., XML, text, HTML). This is determined by the `<xsl:output>` element in the XSLT stylesheet, which defines the format of the resulting file independently of individual templates.

- \* B. Grant access to the XSLT language: This option is nonsensical in the context of XSLT. The `<xsl:template>` element is part of the XSLT language itself and does not "grant access" to it; rather, it is a functional building block used within an XSLT stylesheet.

- \* D. Generate an output file name: The `<xsl:template>` element has no role in naming the output file. In Workday, the output file name is typically configured within the integration system settings (e.g., via the EIB or connector configuration) and is not influenced by the XSLT transformation logic.

An example of `<xsl:template>` in action might look like this in a Workday transformation:

```
<xsl:template match="wd:Worker">
  <Employee>
    <Name><xsl:value-of select="wd:Worker_Name"/></Name>
  </Employee>
</xsl:template>
```

Here, the template matches the Worker node in Workday's XML schema and transforms it into a simpler `<Employee>` structure with a Name element, demonstrating its role in providing rules for node transformation.

References:

- \* Workday Pro Integrations Study Guide: "Configure Integration System - TRANSFORMATION" section, which explains XSLT usage in Workday and highlights `<xsl:template>` as the mechanism for defining transformation rules.

- \* Workday Documentation: "XSLT Transformations in Workday" under the Document Transformation Connector, noting `<xsl:template>` as critical for node-specific processing.

- \* W3C XSLT 1.0 Specification (adopted by Workday): Section 5.3, "Defining Template Rules," which confirms that `<xsl:template>` provides rules for applying transformations to specified nodes.

- \* Workday Community: Examples of XSLT in integration scenarios, consistently using `<xsl:template>` for transformation logic.

## NEW QUESTION # 25

Which three features must all XSLT files contain to be considered valid?

- A. A header, a footer, and a namespace
- **B. A root element, namespace, and at least one template**
- C. A template, a prefix, and a header
- D. A root element, namespace, and at least one transformation

**Answer: B**

Explanation:

For an XSLT (Extensible Stylesheet Language Transformations) file to be considered valid in the context of Workday integrations (and per general XSLT standards), it must adhere to specific structural and functional requirements. The correct answer is that an XSLT file must contain a root element, a namespace, and at least one template. Below is a detailed explanation of why this is the case, grounded in Workday's integration practices and XSLT specifications:

- \* Root Element:

- \* Every valid XSLT file must have a single root element, which serves as the top-level container for the stylesheet. In XSLT, this is

typically the `<xsl:stylesheet>` or `<xsl:transform>` element (both are interchangeable, though `<xsl:stylesheet>` is more common).

- \* The root element defines the structure of the XSLT document and encapsulates all other elements, such as templates and namespaces. Without a root element, the file would not conform to XML well-formedness rules, which are a prerequisite for XSLT validity.

- \* Example:

```
<xsl:stylesheet
version="1.0" xmlns:xsl="http://www.w3.org/1999/XSL/Transform"
>
</xsl:stylesheet>
```

- \* Namespace:

- \* An

XSLT file must declare the XSLT namespace, typically `http://www.w3.org/1999/XSL`

`/Transform`, to identify it as an XSLT stylesheet and enable

the processor to recognize XSLT-specific elements (e.g., `<xsl:template>`, `<xsl:value-of>`). This is declared within the root element using the `xmlns:xsl` attribute.

- \* The namespace ensures that the elements used in the stylesheet are interpreted as XSLT instructions rather than arbitrary XML. Without this namespace, the file would not function as an XSLT stylesheet, as the processor would not know how to process its contents.

- \* In Workday's Document Transformation integrations, additional namespaces (e.g., for Workday-specific schemas) may also be included, but the XSLT namespace is mandatory for validity.

- \* At Least One Template:

- \* An XSLT file must contain at least one `<xsl:template>` element to define the transformation logic. Templates are the core mechanism by which XSLT processes input XML and produces output. They specify rules for matching nodes in the source XML (via the `match` attribute) and generating the transformed result.

- \* Without at least one template, the stylesheet would lack any transformation capability, rendering it functionally invalid for its intended purpose. Even a minimal XSLT file requires a template to produce meaningful output, though built-in default templates exist, they are insufficient for custom transformations like those used in Workday.

- \* Example:

```
<xsl:template match="/">
<result>Hello, Workday!</result>
</xsl:template>
```

Complete Minimal Valid XSLT Example:

```
<xsl:stylesheet
version="1.0" xmlns:xsl="http://www.w3.org/1999/XSL/Transform"
>
<xsl:template match="/">
<output>Transformed Data</output>
</xsl:template>
</xsl:stylesheet>
```

Why Other Options Are Incorrect:

- \* A. A root element, namespace, and at least one transformation: While this is close, "transformation" is not a precise term in XSLT. The correct requirement is a "template," which defines the transformation logic. "Transformation" might imply the overall process, but the specific feature required in the file is a template.

- \* C. A header, a footer, and a namespace: XSLT files do not require a "header" or "footer." These terms are not part of XSLT or XML standards. The structure is defined by the root element and templates, not headers or footers, making this option invalid.

- \* D. A template, a prefix, and a header: While a template is required, "prefix" (likely referring to the namespace prefix like `xsl:`) is not a standalone feature—it's part of the namespace declaration within the root element. "Header" is not a required component, making this option incorrect.

Workday Context:

- \* In Workday's Document Transformation systems (e.g., Core Connectors or custom integrations), XSLT files are uploaded as attachment transformations. Workday enforces these requirements to ensure the stylesheets can process XML data (e.g., from Workday reports or connectors) into formats suitable for external systems. The Workday platform validates these components when an XSLT file is uploaded, rejecting files that lack a root element, namespace, or functional templates.

Workday Pro Integrations Study Guide References:

- \* Workday Integration System Fundamentals: Describes the structure of XSLT files, emphasizing the need for a root element (`<xsl:stylesheet>`), the XSLT namespace, and templates as the building blocks of transformation logic.

- \* Document Transformation Module: Details the requirements for uploading valid XSLT files in Workday, including examples that consistently feature a root element, namespace declaration, and at least one template (e.g., "XSLT Basics for Document Transformation").

- \* Core Connectors and Document Transformation Course Manual: Provides sample XSLT files used in labs, all of which include these three components to ensure functionality within Workday integrations.



\* Workday Community Documentation: Reinforces that XSLT files must be well-formed XML with an XSLT namespace and at least one template to be processed correctly by Workday's integration engine.

### NEW QUESTION # 26

What is the workflow to upload an XSLT file for a brand new Document Transformation system?

- A. Create Integration Attachment Service, then Configure Integration Attachment Service
- B. Configure Integration Attachment Service, then Create Integration Service Attachment
- C. Configure XSLT Attachment Transformation, then Create Integration Attachment Service
- **D. Create XSLT Attachment Transformation, then Configure Integration Attachment Service**

**Answer: D**

Explanation:

In the Workday Pro Integrations program, the process of uploading an XSLT file for a brand-new Document Transformation system follows a specific workflow designed to ensure the transformation logic is properly attached and configured within the integration system. The correct sequence involves first creating the XSLT Attachment Transformation and then configuring the Integration Attachment Service to utilize it. Here's a step-by-step breakdown based on Workday's integration methodology:

\* Create XSLT Attachment Transformation:

\* The initial step is to create an XSLT Attachment Transformation object within Workday. This involves uploading the XSLT file, which contains the transformation logic needed to convert XML data into the desired format for the Document Transformation system. In Workday, XSLT (Extensible Stylesheet Language Transformations) is used to define how data from a source (typically in XML format) is transformed into an output format compatible with an external system.

\* To do this, you navigate to the Integration System, access the related actions, and select the option to create a new "XSLT Attachment Transformation." You then name the transformation, upload the XSLT file (with a size limit of 30 MB as per Workday specifications), and save it.

This step establishes the transformation logic as an object that can be referenced by the integration system.

\* Configure Integration Attachment Service:

\* Once the XSLT Attachment Transformation is created, the next step is to configure the Integration Attachment Service to incorporate this transformation. The Integration Attachment Service is a component of the Document Transformation system that handles the delivery or processing of the transformed data.

\* In this step, you edit the integration system, navigate to the "Services" tab, and configure the Integration Attachment Service. Here, you specify the previously created XSLT Attachment Transformation as the transformation to be applied. This links the XSLT logic to the integration workflow, ensuring that the data processed by the Document Transformation system is transformed according to the uploaded XSLT file.

Why Other Options Are Incorrect:

\* A. Configure XSLT Attachment Transformation, then Create Integration Attachment Service:

This is incorrect because you cannot "configure" an XSLT Attachment Transformation before it exists.

It must first be created as an object in Workday before any configuration or association with services can occur.

\* C. Create Integration Attachment Service, then Configure Integration Attachment Service: This option skips the creation of the XSLT Attachment Transformation entirely, which is a critical step.

Without the transformation defined, configuring the service alone would not enable the XSLT upload or its functionality.

\* D. Configure Integration Attachment Service, then Create Integration Service Attachment: This sequence is reversed and misleading. The Integration Attachment Service must be configured to use an existing XSLT Attachment Transformation, not the other way around. Additionally, "Create Integration Service Attachment" is not a standard term in this context within Workday documentation.

Workday Pro Integrations Study Guide References:

\* Workday Integration System Fundamentals: This section outlines the components of an integration system, including the use of XSLT for document transformation and the role of attachment services.

\* Document Transformation Module: Specifically details the process of uploading and applying XSLT files, emphasizing the creation of an XSLT Attachment Transformation followed by its configuration within the integration services.

\* Core Connectors and Document Transformation Course Manual: Provides practical steps for setting up transformations, including the sequence of creating and then configuring transformation attachments (e.g., Activities related to "Upload a Custom XSLT Transformation" and "Edit XSLT Attachment Transformation").

\* Workday Community Documentation: Confirms that XSLT files are uploaded as attachment transformations and then linked to services like the Integration Attachment Service for processing.

### NEW QUESTION # 27

Refer to the following XML to answer the question below.

```
1. <wd:Get_Job_Profiles_Response xmlns:wd="urn:com.workday/bsvc" wd:version="v43.0">
2.   <wd:Response_Data>
3.     <wd:Job_Profile>
4.       <wd:Job_Profile_Reference>
5.         <wd:ID wd:type="WID">174c31eca2f24ed9b6174ca7d2aeb88c</wd:ID>
6.         <wd:ID wd:type="Job_Profile_ID">Senior_Benefits_Analyst</wd:ID>
7.       </wd:Job_Profile_Reference>
8.       <wd:Job_Profile_Data>
9.         <wd:Job_Code>Senior Benefits Analyst</wd:Job_Code>
10.        <wd:Effective_Date>2024-05-15</wd:Effective_Date>
11.        <wd:Education_Qualification_Replacement_Data>
12.          <wd:Degree_Reference>
13.            <wd:ID wd:type="WID">61383c9b1d094844a73166ad39caebce</wd:ID>
14.            <wd:ID wd:type="Degree_ID">MBA</wd:ID>
15.          </wd:Degree_Reference>
16.          <wd:Field_Of_Study_Reference>
17.            <wd:ID wd:type="WID">62e42dfd4b8c49b5842114f67369a96f</wd:ID>
18.            <wd:ID wd:type="Field_Of_Study_ID">Economics</wd:ID>
19.          </wd:Field_Of_Study_Reference>
20.          <wd:Required>0</wd:Required>
21.        </wd:Education_Qualification_Replacement_Data>
22.      </wd:Education_Qualification_Replacement_Data>
23.      <wd:Degree_Reference>
24.        <wd:ID wd:type="WID">8db9b8e5f53c4cddb7f7a984c6afde28</wd:ID>
25.        <wd:ID wd:type="Degree_ID">B_S</wd:ID>
26.      </wd:Degree_Reference>
27.      <wd:Required>1</wd:Required>
28.    </wd:Education_Qualification_Replacement_Data>
29.  </wd:Job_Profile_Data>
30. </wd:Job_Profile>
31. </wd:Response_Data>
32. </wd:Get_Job_Profiles_Response>
```

You are an integration developer and need to write XSLT to transform the output of an EIB which is making a request to the Get Job Profiles web service operation. The root template of your XSLT matches on the <wd: Get\_Job\_Profiles\_Response> element. This root template then applies a template against <wd:Job\_Profile>.

What XPath syntax would be used to select the value of the wd:Job\_Code element when the <xsl:value-of> element is placed within

the template which matches on <wd:Job\_Profile>?

- A. wd:Job\_Profile\_Data[@wd:Job\_Code]
- B. wd:Job\_Profile/wd:Job\_Profile\_Data/wd:Job\_Code
- C. wd:Job\_Profile\_Reference/wd:ID[@wd:type='Job\_Profile\_ID']
- D. wd:Job\_Profile\_Data/wd:Job\_Code

**Answer: D**

**Explanation:**

As an integration developer working with Workday, you are tasked with transforming the output of an Enterprise Interface Builder (EIB) that calls the Get\_Job\_Profiles web service operation. The provided XML shows the response from this operation, and you need to write XSLT to select the value of the <wd: Job\_Code> element.

The root template of your XSLT matches on <wd: Get\_Job\_Profiles\_Response> and applies a template to <wd:Job\_Profile>. Within this template, you use the <xsl:value-of> element to extract the <wd:Job\_Code> value. Let's analyze the XML structure, the requirement, and each option to determine the correct XPath syntax.

**Understanding the XML and Requirement**

The XML snippet provided is a SOAP response from the Get\_Job\_Profiles web service operation in Workday, using the namespace xmlns:wd="urn:com.workday/bsvc" and version wd:version="v43.0". Key elements relevant to the question include:

- \* The root element is <wd: Get\_Job\_Profiles\_Response>.
- \* It contains <wd:Response\_Data>, which includes <wd:Job\_Profile> elements.
- \* Within <wd:Job\_Profile>, there are:
  - \* <wd:Job\_Profile\_Reference>, which contains <wd:ID> elements (e.g., a Job\_Profile\_ID).
  - \* <wd:Job\_Profile\_Data>, which contains <wd:Job\_Code> with the value Senior\_Benefits\_Analyst.

The task is to select the value of <wd:Job\_Code> (e.g., "Senior\_Benefits\_Analyst") using XPath within an XSLT template that matches <wd:Job\_Profile>. The <xsl:value-of> element outputs the value of the selected node, so you need the correct XPath path from the <wd:Job\_Profile> context to <wd:Job\_Code>.

**Analysis of Options**

Let's evaluate each option based on the XML structure and XPath syntax rules:

- \* Option A: wd:Job\_Profile/wd:Job\_Profile\_Data/wd:Job\_Code

\* This XPath starts from `wd:Job_Profile` and navigates to `wd:Job_Profile_Data/wd:Job_Code`.

However, in the XML, `<wd:Job_Profile>` is the parent element, and `<wd:Job_Profile_Data>` is a direct child containing `<wd:Job_Code>`. The path `wd:Job_Profile/wd:Job_Profile_Data/wd:`

`Job_Code` is technically correct in terms of structure, as it follows the hierarchy:

\* `<wd:Job_Profile> # <wd:Job_Profile_Data> # <wd:Job_Code>`.

\* However, since the template matches `<wd:Job_Profile>`, the context node is already `<wd:`

`Job_Profile>`. You don't need to include `wd:Job_Profile/` at the beginning of the XPath unless navigating from a higher level. Starting directly with `wd:Job_Profile_Data/wd:Job_Code` (Option C) is more concise and appropriate for the context. This option is technically valid but redundant and less efficient, making it less preferred compared to Option C.

\* Option B: `wd:Job_Profile_Data[@wd:Job_Code]`

\* This XPath uses an attribute selector (`[@wd:Job_Code]`) to filter `<wd:Job_Profile_Data>` based on an attribute named `wd:Job_Code`. However, examining the XML, `<wd:Job_Profile_Data>` does not have a `wd:Job_Code` attribute—it has a child element `<wd:Job_Code>` with the value

"Senior\_Benefits\_Analyst." The `[@attribute]` syntax is used for attributes, not child elements, so this XPath is incorrect. It would not select the `<wd:Job_Code>` value and would likely return no results or an error. This option is invalid.

\* Option C: `wd:Job_Profile_Data/wd:Job_Code`

\* This XPath starts from `wd:Job_Profile_Data` (a direct child of `<wd:Job_Profile>`) and navigates to `wd:Job_Code`. Since the template matches `<wd:Job_Profile>`, the contextnode is `<wd:`

`Job_Profile>`, and `wd:Job_Profile_Data/wd:Job_Code` correctly points to the `<wd:Job_Code>` element within `<wd:Job_Profile_Data>`. This path is:

\* Concise and appropriate for the context.

\* Directly selects the value "Senior\_Benefits\_Analyst" when used with `<xsl:value-of>`.

\* Matches the XML structure, as `<wd:Job_Profile_Data>` contains `<wd:Job_Code>` as a child.

\* This is the most straightforward and correct option for selecting the `<wd:Job_Code>` value within the `<wd:Job_Profile>` template.

\* Option D: `wd:Job_Profile_Reference/wd:ID[@wd:type='Job_Profile_ID']`

\* This XPath navigates to `<wd:Job_Profile_Reference>` (a child of `<wd:Job_Profile>`) and then to

`<wd:ID>` with an attribute `wd:type='Job_Profile_ID'`. In the XML, `<wd:Job_Profile_Reference>` contains:

\* `<wd:ID wd:type="WID">1740d3eca2f2ed9b6174ca7d2ae88c8c</wd:ID>`

\* `<wd:ID wd:type="Job_Profile_ID">Senior_Benefits_Analyst</wd:ID>`

\* The XPath `wd:Job_Profile_Reference/wd:ID[@wd:type='Job_Profile_ID']` selects the `<wd:ID>` element with `wd:type='Job_Profile_ID'`, which has the value "Senior\_Benefits\_Analyst." However, this is not the `<wd:Job_Code>` value—the `<wd:Job_Code>` is a separate element under

`<wd:Job_Profile_Data>`, not `<wd:Job_Profile_Reference>`. The question specifically asks for the

`<wd:Job_Code>` value, so this option is incorrect, as it selects a different piece of data (the job profile ID, not the job code).

Why Option C is Correct

Option C, `wd:Job_Profile_Data/wd:Job_Code`, is the correct XPath syntax because:

\* It starts from the context node `<wd:Job_Profile>` (as the template matches this element) and navigates to

`<wd:Job_Profile_Data/wd:Job_Code>`, which directly selects the `<wd:Job_Code>` element's value ("Senior\_Benefits\_Analyst").

\* It is concise and aligns with standard XPath navigation in XSLT, avoiding unnecessary redundancy (unlike Option A) or incorrect attribute selectors (unlike Option B).

\* It matches the XML structure, where `<wd:Job_Profile_Data>` is a child of `<wd:Job_Profile>` and contains `<wd:Job_Code>` as a child.

\* When used with `<xsl:value-of select="wd:Job_Profile_Data/wd:Job_Code"/>` in the template, it outputs the job code value, fulfilling the requirement.

Practical Example in XSLT

Here's how this might look in your XSLT:

xml

WrapCopy

```
<xsl:template match="wd:Job_Profile">
```

```
<xsl:value-of select="wd:Job_Profile_Data/wd:Job_Code"/>
```

```
</xsl:template>
```

This would output "Senior\_Benefits\_Analyst" for the `<wd:Job_Code>` element in the XML.

Verification with Workday Documentation

The Workday Pro Integrations Study Guide and SOAP API Reference (available via Workday Community) detail the structure of the `Get_Job_Profiles` response and how to use XPath in XSLT for transformations. The XML structure shows

`<wd:Job_Profile_Data>` as the container for job profile details, including `<wd:`

`Job_Code>`. The guide emphasizes using relative XPath paths within templates to navigate from the matched element (e.g.,

`<wd:Job_Profile>`) to child elements like `<wd:Job_Profile_Data/wd:Job_Code>`.

Workday Pro Integrations Study Guide References

\* Section: XSLT Transformations in EIBs- Describes using XSLT to transform web service responses, including selecting elements with XPath.



\* Section: Workday Web Services- Details the Get\_Job\_Profiles operation and its XML output structure, including <wd:Job\_Profile\_Data> and <wd:Job\_Code>.

\* Section: XPath Syntax- Explains how to navigate XML hierarchies in Workday XSLT, using relative paths like wd:Job\_Profile\_Data/wd:Job\_Code from a <wd:Job\_Profile> context.

\* Workday Community SOAP API Reference - Provides examples of XPath navigation for Workday web service responses. Option C is the verified answer, as it correctly selects the <wd:Job\_Code> value using the appropriate XPath syntax within the <wd:Job\_Profile> template context.

## NEW QUESTION # 28

.....

As for candidates who will attend the exam, choosing the practicing materials may be a difficult choice. Then just trying Workday-Pro-Integrations learning materials of us, with the pass rate is 98.95%, we help the candidates to pass the exam successfully. Many candidates have sent their thanks to us for helping them to pass the exam by using the Workday-Pro-Integrations Learning Materials. The reason why we gain popularity in the customers is the high-quality of Workday-Pro-Integrations exam dumps. In addition, we provide you with free update for one year after purchasing. Our system will send the latest version to you email address automatically.

**Workday-Pro-Integrations Exam Outline:** <https://www.itexamdownload.com/Workday-Pro-Integrations-valid-questions.html>

No matter which method you choose, as long as you ask for Workday-Pro-Integrations learning materials, we guarantee that we will reply to you as quickly as possible, The ITExamDownload offers real, valid, and updated Workday-Pro-Integrations Questions that surely will help you in exam preparation and enable you to pass the challenging Workday-Pro-Integrations exam with flying colors, Workday Latest Study Workday-Pro-Integrations Questions They can even broaden amplitude of your horizon in this line.

Practice those two, and the rest will follow, Managing Your Browser Window, No matter which method you choose, as long as you ask for Workday-Pro-Integrations learning materials, we guarantee that we will reply to you as quickly as possible.

## Why Choose ITExamDownload for Workday Workday-Pro-Integrations Exam Questions Preparation?

The ITExamDownload offers real, valid, and updated Workday-Pro-Integrations Questions that surely will help you in exam preparation and enable you to pass the challenging Workday-Pro-Integrations exam with flying colors.

They can even broaden amplitude of your horizon in this line, Our Workday-Pro-Integrations study materials not only target but also cover all knowledge points, ITExamDownload have the obligation to ensure your comfortable learning if you have spent money on our Workday-Pro-Integrations study materials.

- Get Reliable Latest Study Workday-Pro-Integrations Questions and Pass Exam in First Attempt ☐ Search for ► Workday-Pro-Integrations ◀ on ► [www.real4dumps.com](http://www.real4dumps.com) ☐ immediately to obtain a free download ☐ Valid Dumps Workday-Pro-Integrations Sheet
- Simulated Workday-Pro-Integrations Test ☐ Latest Workday-Pro-Integrations Test Preparation ☐ Related Workday-Pro-Integrations Exams ☐ Download ➡ Workday-Pro-Integrations ☐ for free by simply entering { [www.pdfvce.com](http://www.pdfvce.com) } website ☐ Reliable Workday-Pro-Integrations Study Guide
- Simulated Workday-Pro-Integrations Test ☐ Technical Workday-Pro-Integrations Training ☐ Related Workday-Pro-Integrations Exams ☐ Easily obtain ( Workday-Pro-Integrations ) for free download through ➡ [www.vceengine.com](http://www.vceengine.com) ☐ ☐ Reliable Workday-Pro-Integrations Practice Questions
- Get Reliable Latest Study Workday-Pro-Integrations Questions and Pass Exam in First Attempt ☐ Enter ➡ [www.pdfvce.com](http://www.pdfvce.com) ☐ and search for 《 Workday-Pro-Integrations 》 to download for free ☐ Workday-Pro-Integrations Test Tutorials
- Reliable Workday-Pro-Integrations Study Guide ☒ Valid Dumps Workday-Pro-Integrations Sheet ☐ Test Workday-Pro-Integrations Sample Online ☐ Search for ► Workday-Pro-Integrations ◀ and download exam materials for free through ► [www.prep4pass.com](http://www.prep4pass.com) ◀ ☐ Test Workday-Pro-Integrations Sample Online
- Workday-Pro-Integrations Reliable Exam Online ☐ Free Workday-Pro-Integrations Practice ☐ Free Workday-Pro-Integrations Practice ☐ Simply search for { Workday-Pro-Integrations } for free download on 「 [www.pdfvce.com](http://www.pdfvce.com) 」 ☐ ☐ Reliable Workday-Pro-Integrations Exam Review
- Valid Workday-Pro-Integrations Study Notes ☐ Workday-Pro-Integrations Exam Testking ☐ Technical Workday-Pro-Integrations Training ☐ 《 [www.vceengine.com](http://www.vceengine.com) 》 is best website to obtain [ Workday-Pro-Integrations ] for free download ☐ Valid Workday-Pro-Integrations Study Notes
- 2025 Latest Study Workday-Pro-Integrations Questions | Reliable 100% Free Workday Pro Integrations Certification Exam

Exam Outline □ Search for **【 Workday-Pro-Integrations 】** and download exam materials for free through ► [www.pdfvce.com](http://www.pdfvce.com) ◀ □ Workday-Pro-Integrations Customizable Exam Mode

- Valid Workday-Pro-Integrations Study Notes □ Test Workday-Pro-Integrations Sample Online 📄 Test Workday-Pro-Integrations Sample Online □ The page for free download of “Workday-Pro-Integrations ” on ► [www.prep4away.com](http://www.prep4away.com) ◀ will open immediately □ Workday-Pro-Integrations Exam Simulator Free
- Simulated Workday-Pro-Integrations Test □ Workday-Pro-Integrations Exam Testking □ Valid Workday-Pro-Integrations Study Notes □ Download □ Workday-Pro-Integrations □ for free by simply searching on ► [www.pdfvce.com](http://www.pdfvce.com) □ □ Workday-Pro-Integrations Customizable Exam Mode
- 2025 Latest Study Workday-Pro-Integrations Questions | Reliable 100% Free Workday Pro Integrations Certification Exam Exam Outline □ Search for ► Workday-Pro-Integrations □□□ on ( [www.actual4labs.com](http://www.actual4labs.com) ) immediately to obtain a free download □ Related Workday-Pro-Integrations Exams
- [www.stes.tyc.edu.tw](http://www.stes.tyc.edu.tw), [myportal.utt.edu.tt](http://myportal.utt.edu.tt), [myportal.utt.edu.tt](http://myportal.utt.edu.tt), [myportal.utt.edu.tt](http://myportal.utt.edu.tt), [myportal.utt.edu.tt](http://myportal.utt.edu.tt), [myportal.utt.edu.tt](http://myportal.utt.edu.tt), [myportal.utt.edu.tt](http://myportal.utt.edu.tt), [myportal.utt.edu.tt](http://myportal.utt.edu.tt), [myportal.utt.edu.tt](http://myportal.utt.edu.tt), [myportal.utt.edu.tt](http://myportal.utt.edu.tt), [myportal.utt.edu.tt](http://myportal.utt.edu.tt), [study.stcs.edu.np](http://study.stcs.edu.np), [motionentrance.edu.np](http://motionentrance.edu.np), [www.stes.tyc.edu.tw](http://www.stes.tyc.edu.tw), [www.stes.tyc.edu.tw](http://www.stes.tyc.edu.tw), [www.stes.tyc.edu.tw](http://www.stes.tyc.edu.tw), [www.stes.tyc.edu.tw](http://www.stes.tyc.edu.tw), [motionentrance.edu.np](http://motionentrance.edu.np), [adleading.com](http://adleading.com), [studyduke.inkliksites.com](http://studyduke.inkliksites.com), Disposable vapes

BTW, DOWNLOAD part of ITExamDownload Workday-Pro-Integrations dumps from Cloud Storage:  
[https://drive.google.com/open?id=1pF9Km-Q1cywqdLnS2e9ulk\\_e74Ipr\\_C](https://drive.google.com/open?id=1pF9Km-Q1cywqdLnS2e9ulk_e74Ipr_C)