Workday Workday-Pro-Integrations Testfagen & Workday-Pro-Integrations Testantworten

P.S. Kostenlose 2026 Workday Workday-Pro-Integrations Prüfungsfragen sind auf Google Drive freigegeben von DeutschPrüfung verfügbar: https://drive.google.com/open?id=1ajViXpG4325g4xaL25hwo4tLdR3q_VgF

Um jeder Workday Workday-Pro-Integrations Prüfungsunterlagen Benutzer einen bequemen Prozess zu haben, bieten wir Ihnen 3 Versionen von Workday Workday-Pro-Integrations Prüfungsunterlagen, nämlich PDF-, Online-, und Software-Version. Eine der Versionen kann für Sie taugen und Ihnen helfen, innerhalb der kürzesten Zeit Workday Workday-Pro-Integrations zu bestehen und die autoritativste internationale Zertifizierung zu erwerben!

Workday Workday-Pro-Integrations Exam Syllabus Topics:

Section	Objectives
Reporting	- Building, modifying, and managing Workday reports for integrations • 1. Custom report types • 2. Optimizing report performance for automated data exchange • 3. Working with report writer tools • 4. Calculated fields within reports
Integrations	- Core integration architecture • 1. Integration system user setup • 2. Workday Studio • 3. APIs - Building scalable, maintainable, and secure integrations • 1. Ensuring seamless system interoperability

Cloud Connect	- Using Workday Cloud Connect solutions for third-party integration • 1. Configuration settings • 2. Pre-built connectors • 3. Managing data flow between Workday and external systems • 4. Ensuring security and data integrity
Calculated Fields	- Creation, configuration, and management of calculated fields • 1. Transforming, manipulating, and formatting data • 2. Understanding field types, dependencies, and logical operations • 3. Dynamic data customization within integration workflows
XSLT	- Transforming XML data structures • 1. Formatting output for integration use cases (APIs, external file delivery) • 2. Applying conditional logic

>> Workday Workday-Pro-Integrations Testfagen <<

Workday-Pro-Integrations Braindumpsit Dumps PDF & Workday Workday-Pro-Integrations Braindumpsit IT-Zertifizierung - Testking Examen Dumps

Bemühen Sie sich noch um die Workday Workday-Pro-Integrations Zertifizierungsprüfung? Wollen Sie schneller Ihren Traum verwirklichen? Bitte wählen Sie die Workday-Pro-Integrations Schulungsmaterialien von DeutschPrüfung. Wenn Sie DeutschPrüfung wählen, ist es kein Traum mehr, das Workday Workday-Pro-Integrations Zertifikat zu erhalten.

Workday Pro Integrations Certification Exam Workday-Pro-Integrations Prüfungsfragen mit Lösungen (Q67-Q72):

67. Frage
Refer to the following XML and example transformed output to answer the question below.

```
<wd:Report_Data xmlns:wd="urn:com.workday.report/Int_Report">
    <wd:Report_Entry>
        <wd:Worker>Logan McNeil</wd:Worker>
        <wd:Education_Group>
            <wd:Education>California University</wd:Education>
            <wd:Degree>MBA</wd:Degree>
        </wd:Education_Group>
        <wd:Education_Group>
            <wd:Education>Georgetown University</wd:Education>
            <wd:Degree>B.S.</wd:Degree>
        </wd:Education_Group>
    </wd:Report_Entry>
    <wd:Report_Entry>
        <wd:Worker>Steve Mewman</wd:Worker>
        <wd:Education_Group>
            <wd:Education>Iowa State University</wd:Education>
            <wd:Degree>B.A.</wd:Degree>
        </wd:Education_Group>
        <wd:Education_Group>
            <wd:Education>Northwestern University</wd:Education>
            <wd:Degree>MBA</wd:Degree>
        </wd:Education_Group>
    </wd:Report_Entry>
</wd:Report_Data>
```


Example transformed wd:Report_Entry output;

```
1. <Transformed_Record>
2.     <Worker>Logan McNeil</Worker>
3.     <Degrees>
4.         <Degree>California University MBA</Degree>
5.         <Degree>Georgetown University B.S.</Degree>
6.     </Degrees>
7. </Transformed_Record>
```

What is the XSLT syntax tor a template that matches onwd: Educationj3roup to produce the degree data in the above Transformed_Record example?

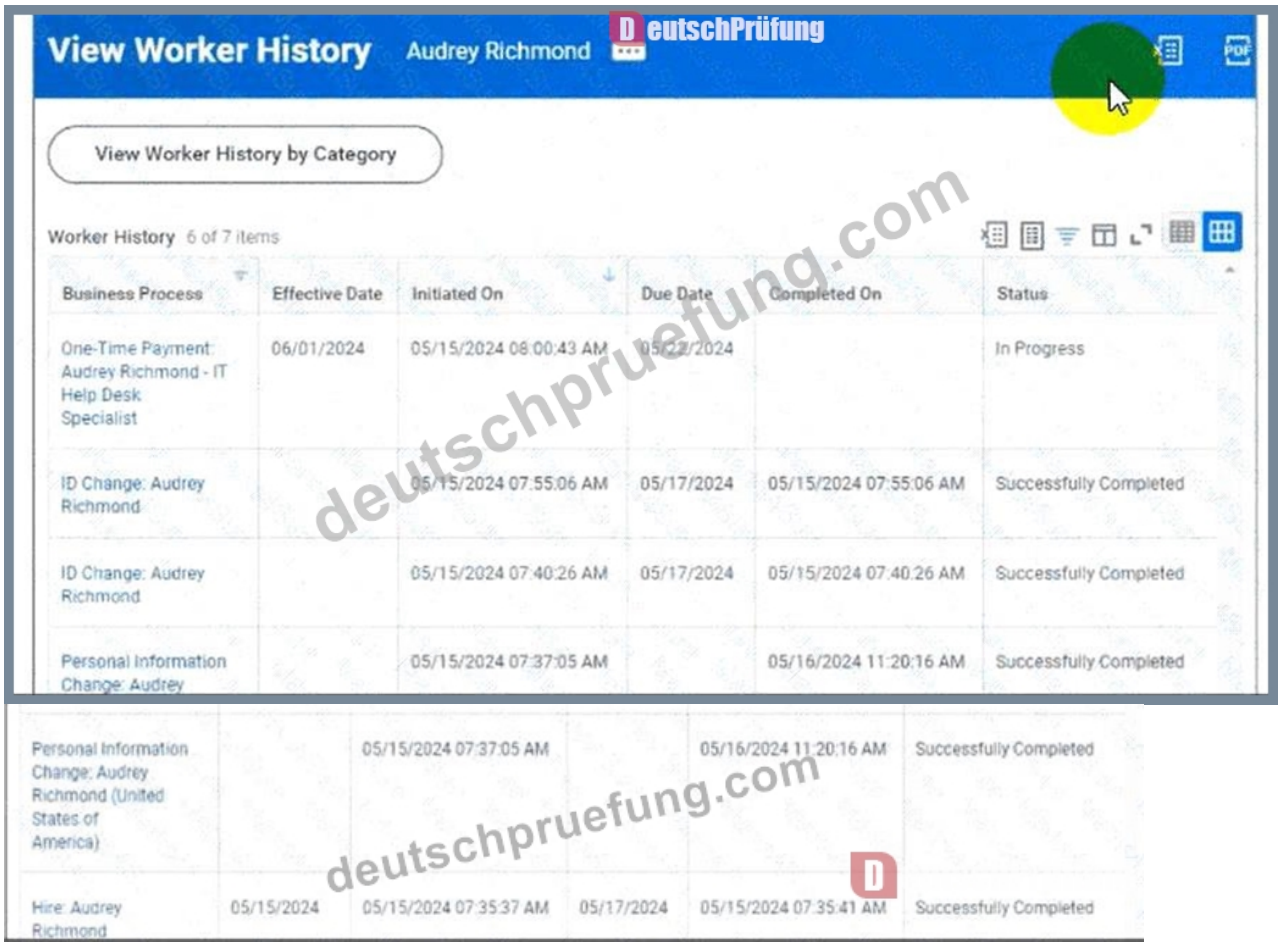
- A.
```
1. <xsl:template match="wd:Education_Group">
2.     <Degree>
3.         <xsl:value-of select="*"/>
4.     </Degree>
5. </xsl:template>
```
- B.
```
1. <xsl:template match="wd:Education_Group">
2.     <Degree>
3.         <xsl:copy><xsl:value-of select="*"/></xsl:copy>
4.     </Degree>
5. </xsl:template>
```
```
1. <xsl:template match="wd:Education_Group">
2.     <Degree>
3.         <xsl:copy-of select="*"/>
4.     </Degree>
5. </xsl:template>
```
- C.
- D.
```
1. <xsl:template match="wd:Education_Group">
2.     <Degree>
3.         <xsl:copy select="*"/>
4.     </Degree>
5. </xsl:template>
```

Antwort: B

Begründung:
In Workday integrations, XSLT is used to transform XML data, such as the output from a web service- enabled report or EIB, into a desired format for third-party systems. In this scenario, you need to create an XSLT template that matches the wd:Education_Group element in the provided XML and transforms it to produce the degree data in the format shown in the Transformed_Record example. The goal is to output each degree (e.g., "California University MBA" and "Georgetown University B.S.") as a <Degree> element within a <Degrees> parent element.
Here's why option A is correct:
* Template Matching: The <xsl:template match="wd:Education_Group"> correctly targets the wd:
Education_Group element in the XML, which contains multiple wd:Education elements, each with a wd:Degree child, as shown in the XML snippet (e.g., <wd:Education>California University</wd:
Education><wd:Degree>MBA</wd:Degree>).
* Transformation Logic:
* <Degree> creates the outer <Degree> element for each education group, matching the structure in the Transformed_Record example (e.g., <Degree>California University MBA</Degree>).
* <xsl:copy><xsl:value-of select="*"/></xsl:copy> copies the content of the child elements (wd:
Education and wd:Degree) and concatenates their values into a single string. The select="*" targets all child elements of wd:Education_Group, and xsl:value-of outputs their text content (e.
g., "California University" and "MBA" become "California University MBA").
* This approach ensures that each wd:Education_Group is transformed into a single <Degree> element with the combined text of the wd:Education and wd:Degree values, matching the example output.
* Context and Output: The template operates on each wd:Education_Group, producing the nested structure shown in the Transformed_Record (e.g., <Degrees><Degree>CaliforniaUniversity MBA<

/Degree><Degree>Georgetown University B.S.</Degree></Degrees>), assuming a parent template or additional logic wraps the <Degree> elements in <Degrees>.
Why not the other options?
* B.
xml
WrapCopy
<xsl:template match="wd:Education_Group">
<Degree>
<xsl:value-of select="*"/>
</Degree>
</xsl:template>
This uses <xsl:value-of select="*"/> without <xsl:copy>, which outputs the concatenated text of all child elements but does not preserve any XML structure or formatting. It would produce plain text (e.g., "California UniversityMBACalifornia UniversityB.S.") without the proper <Degree> tags, failing to match the structured output in the example.
* C.
xml
WrapCopy
<xsl:template match="wd:Education_Group">
<Degree>
<xsl:copy select="*"/>
</Degree>
</xsl:template>
This uses <xsl:copy select="*"/>, but <xsl:copy> does not take a select attribute-it simply copies the current node. This would result in an invalid XSLT syntax and fail to produce the desired output, making it incorrect.
* D.
xml
WrapCopy
<xsl:template match="wd:Education_Group">
<Degree>
<xsl:copy-of select="*"/>
</Degree>
</xsl:template>
This uses <xsl:copy-of select="*"/>, which copies all child nodes (e.g., wd:Education and wd:Degree) as-is, including their element structure, resulting in output like <Degree><wd:Education>California University</wd:
Education><wd:Degree>MBA</wd:Degree></Degree>. This does not match the flattened, concatenated text format in the Transformed_Record example (e.g., <Degree>California University MBA</Degree>), making it incorrect.
To implement this in XSLT for a Workday integration:
* Use the template from option A to match wd:Education_Group, apply <xsl:copy><xsl:value-of select="
*"/></xsl:copy> to concatenate and output the wd:Education and wd:Degree values as a single
<Degree> element. This ensures the transformation aligns with the Transformed_Record example, producing the required format for the integration output.
References:
* Workday Pro Integrations Study Guide: Section on "XSLT Transformations for Workday Integrations"
- Details the use of <xsl:template>, <xsl:copy>, and <xsl:value-of> for transforming XML data, including handling grouped elements like wd:Education_Group.
* Workday EIB and Web Services Guide: Chapter on "XML and XSLT for Report Data" - Explains the structure of Workday XML (e.g., wd:Education_Group, wd:Education, wd:Degree) and how to use XSLT to transform education data into a flattened format.
* Workday Reporting and Analytics Guide: Section on "Web Service-Enabled Reports" - Covers integrating report outputs with XSLT for transformations, including examples of concatenating and restructuring data for third-party systems.

68. Frage
Refer to the following scenario to answer the question below.
A connector is configured to detect changes to government IDs, personal information, and compensation data.
The worker history report below shows recent transactions for a new hire.



Worker History: Audrey Richmond
The worker history includes a One Time Payment transaction for Audrey Richmond with an effective date of 05/01/2024, initiated on 05/15/2024, due date of 05/17/2024, and status of In Progress. Other transactions shown include ID Change, Personal Information Change, Hire, and Propose Compensation records.
You launch the connector integration to process changes with the following parameters:
As Of Entry Moment: 05/17/2024 12:00:00 AM
Effective Date: 05/17/2024
Last Successful As Of Entry Moment: 05/15/2024 12:00:00 AM
Last Successful Effective Date: 05/15/2024
Why will the integration output file exclude the one-time payment change?

- A. An HR administrator rescinded the change.
- B. Effective date is outside the launch timeline.
- C. Status of the change is in progress.
- D. Due date is outside the launch timeline.

Antwort: C

Begründung:
Connector change detection relies on completed business process transactions that are eligible for extraction within the launch parameters. In the worker history, the One Time Payment transaction is still marked as In Progress. An in-progress transaction has not completed the business process lifecycle, so it is not treated as a finalized change ready for outbound integration processing. The due date is not the controlling reason, and there is no indication that the transaction was rescinded by an HR administrator. Although effective dates matter in connector processing, the most direct exclusion reason shown in the worker history is the transaction status. Workday integrations generally avoid sending incomplete business process results to external systems because doing so could transmit unapproved or unfinished compensation data.

69. Frage
Refer to the scenario. You are configuring a Core Connector: Worker integration with the Data Initialization Service (DIS) enabled. The integration must extract worker contact details and job information, including a calculated field override that determines phone

allowance eligibility.
While testing, the output contains no records, and the Messages tab shows exception logs stating you don't have access to the Exempt field. You note this is the same field being used for Population Eligibility in the integration.
What must you configure to resolve this security issue?

- A. Assign the ISSG to a row with Modify access in the domain security policy securing the Population Eligibility field.
- B. Assign the ISSG to a row with View access in the domain security policy securing the Web Service.
- C. Assign the ISSG to a row with Modify access in the domain security policy securing the Web Service.
- D. Assign the ISSG to a row with View access in the domain security policy securing the Population Eligibility field.

Antwort: D

Begründung:
The Exempt field is being used in Population Eligibility, and eligibility fields must be readable by the ISSG. If the domain security policy for a field denies View access, Workday cannot evaluate the eligibility and returns no data.
From Workday security governance:
"For integrations using Population Eligibility, the ISSG must have View permission on all fields referenced in eligibility rules." If View is missing, the eligibility rule cannot execute → No workers are considered eligible → Output contains zero records → Error logged for denied field access.
Therefore, the solution is:
* Grant the ISSG View access to the domain that secures the Population Eligibility field Modify access (A/C) is not needed - eligibility only needs read-access.

70. Frage
What XSL component is required to execute valid transformation instructions in the XSLT code?

- A. xsl:output
- B. xsl:apply-template
- C. xsl:template
- D. xsl:call-template

Antwort: C

Begründung:
The <xsl:template> is the core component in XSLT. It defines the transformation rules that will be applied to nodes in the XML document.
"Without at least one <xsl:template> element, an XSLT file cannot perform any transformation. This is the execution block where processing logic begins." Why the others are incorrect:
* B. <xsl:apply-templates> applies templates but is not valid without the actual template definitions.
* C. <xsl:call-template> calls named templates - which must first exist.
* D. <xsl:output> defines format but does not perform transformation logic.
Reference:W3C XSLT Specification - Section: xsl:template Required for ExecutionWorkday XSLT Examples - "Template-Based Transformations in Workday"

71. Frage
A vendor needs to create a Date Difference calculated field. However, the two dates needed for that calculation are on two separate business objects.
What additional calculated field do you need to create that Date Difference calculated field?

- A. Lookup Related Value
- B. Build Date
- C. Lookup Value as of Date
- D. Lookup Date Rollup

Antwort: A

Begründung:
When creating a Date Difference calculated field in Workday, both dates must exist on the same business object. If they are on different business objects, you need to first bring the second date onto the primary object. To do that, you use a:

Lookup Related Value calculated field - this allows you to retrieve a field (like a date) from a related business object, so it can then be used in further calculations.
Example scenario:
* You want to subtract Hire Date (on the Worker object) from Dependent's Birth Date (on the Dependent object).
* These are on different objects # use Lookup Related Value to pull the second date into the current object context.
* Then, create the Date Difference using both dates on the same object.
Why other options are incorrect:
* B. Build Date creates a synthetic date, not for bridging objects.
* C. Lookup Date Rollup rolls up values across multiple related objects, not typically used for 1-to-1 value bridging.
* D. Lookup Value as of Date is used for time-sensitive lookups (e.g., point-in-time values), not structural bridging.
Reference:Workday Pro: Calculated Fields - Working Across Business Objects with Lookup Related ValueWorkday Community: Bringing Dates Across Objects to Support Date Difference Calculations

72. Frage
.....

Die Produkte von DeutschPrüfung sind von guter Qualität. Sie sind am schnellsten aktualisiert. Wenn Sie die Schulungsunterlagen zur Workday Workday-Pro-Integrations Zertifizierungsprüfung kaufen, können Sie die Workday Workday-Pro-Integrations Zertifizierungsprüfung sicher bestehen.

myportal.utt.edu.tt, myportal.utt.edu.tt, myportal.utt.edu.tt, myportal.utt.edu.tt, myportal.utt.edu.tt, myportal.utt.edu.tt, myportal.utt.edu.tt, www.stes.tyc.edu.tw, www.stes.tyc.edu.tw, www.stes.tyc.edu.tw, www.stes.tyc.edu.tw, Disposable vapes

P.S. Kostenlose und neue Workday-Pro-Integrations Prüfungsfragen sind auf Google Drive freigegeben von DeutschPrüfung verfügbar: https://drive.google.com/open?id=1ajViXpG4325g4xaL25hwo4tLdR3q_VgF