

## [100% Pass Unlimited Download Passleader Free Microsoft 70-464 Exam Practice Test (91-110)]

Get New Valid Dumps To Pass Exam 70-464: The following new 70-464 exam questions were updated in recent days by PassLeader, visit [passleader.com](#) to get the full version of new 70-464 153q exam dumps with free version of new VCE Player software, our valid 70-464 153q briandump will help you passing 70-464 exam easily! keywords: 70-464 exam,70-464 exam dumps,70-464 153q exam questions,70-464 pdf dumps,70-464 vce dumps,70-464 153q briandump,Developing Microsoft SQL Server 2014 Databases Exam

Compare And Choose The Best PassLeader 70-464 Brain Dumps

Pass4sure	PL PassLeader	TEST KING
Banned By Microsoft Not Available	153 Q&As Price: \$99.99 Coupon Code -- CELEB	50 Q&As Price: \$124.99

QUESTION 91 You need to recommend a solution that meets the concurrency problems. What should you include in the recommendation? A. Modify the stored procedures to use the SERIALIZABLE isolation level. B. Modify the order in which usp\_UpdateCandidate accesses the Applications table and the Candidates table. C. Modify the stored procedures to use the REPEATABLE READ isolation level. D. Modify the order in which usp\_AcceptCandidate accesses the Applications table and the Candidates table.

Answer: B QUESTION 92 You need to implement a change to usp\_ExportOpenings that meets the integration requirements. What should you modify in usp\_ExportOpenings? (Each correct answer presents part of the solution. Choose all that apply?)

A. To the end of line 04, add [Opening]. B. To the end of line 05, add [Opening!title]. C. To line 10, add FOR XML RAW. D. To line 10, add FOR XML EXPLICIT. E. To line 10, add FOR XML AUTO. F. To the end of line 04, add [Opening!ELEMENT]. G. To the end of line 06, add [Opening!salary!ELEMENT]. H. To the end of line 05, add [Opening!title!ELEMENT]. I. To the end of line 06, add [Opening! salary]. Answer: ABEI QUESTION 93 You need to implement a solution that addresses the upload requirements. Which code segment should you use to implement the Conversions assembly?

- A. 

```
CREATE FUNCTION ConvertToText (@wordResume varchar(8000))
RETURNS varchar(8000)
AS EXTERNAL NAME Conversions.SqlConversions.ConvertToText
```
- B. 

```
CREATE FUNCTION ConvertToText (@wordResume varchar(8000))
RETURNS varbinary(max)
AS EXTERNAL NAME Conversions.SqlConversions.ConvertToText
```
- C. 

```
CREATE PROCEDURE ConvertToText (@wordResume varchar(8000))
AS EXTERNAL NAME Conversions.SqlConversions.ConvertToText
```
- D. 

```
CREATE PROCEDURE ConvertToText (@wordResume varchar(8000))
AS EXTERNAL NAME Conversions.SqlConversions.ConvertToText
```

A. Option A B. Option B C. Option C D. Option D Answer: A QUESTION 94 You need to implement a solution that meets the locking requirements. Which line of code should you modify? A. Change line 07 in usp\_UpdateOpening to: UPDATE Openings WITH (UPDLOCK) B. Change line 09 in usp\_GetOpenings to: FROM Openings o (ROWLOCK) C. Change line 07 in usp\_UpdateOpening to: UPDATE Openings WITH (READPAST) D. Change line 09 in usp\_GetOpenings to: FROM Openings o (NOLOCK) Answer: D QUESTION 95 You need to implement a solution that meets the security requirements. Which statement should you execute?

- A. REVOKE EXEC ON usp\_UpdateOpening FROM Candidates;
- B. DENY EXEC ON usp\_UpdateOpening TO Candidates;
- C. ALTER PROCEDURE usp\_UpdateOpening  
@openingIDint,  
@titlevarchar(100),  
@salarydecimal(18,0),  
@descriptionvarchar(8000)  
WITH EXECUTE AS Administrator  
AS  
...
- D. ALTER PROCEDURE usp\_UpdateOpening  
@openingIDint,  
@titlevarchar(100),  
@salarydecimal(18,0),  
@descriptionvarchar(8000)  
WITH EXECUTE AS Company  
AS  
...

A.&#160;&#160;&#160; Option A B.&#160;&#160;&#160; Option B C.&#160;&#160;&#160; Option C  
D.&#160;&#160;&#160; Option D Answer: A QUESTION 96 You need to implement a solution that resolves the salary query issue. Which statement should you execute on DB1?

- A. UPDATE Openings SET Salary=0 WHERE Salary IS NULL;  
GO  
ALTER TABLE Openings  
WITH NOCHECK  
MODIFY COLUMN Salary NOT NULL;  
GO  
ALTER TABLE Openings  
WITH NOCHECK  
ADD CONSTRAINT DF\_SALARY  
DEFAULT 0 FOR Salary;  
GO
- B. ALTER TABLE Openings  
WITH NOCHECK  
ADD CONSTRAINT DF\_SALARY  
DEFAULT 0 FOR Salary;  
GO  
ALTER TABLE Openings  
WITH NOCHECK  
MODIFY COLUMN Salary NULL;  
GO  
UPDATE Openings SET Salary=0 WHERE Salary IS NULL;  
GO
- C. UPDATE Openings SET Salary=0 WHERE Salary IS NULL;  
GO  
ALTER TABLE Openings  
WITH NOCHECK  
ADD CONSTRAINT CT\_SALARY  
CHECK (Salary>=0);  
GO  
ALTER TABLE Openings  
WITH NOCHECK  
MODIFY COLUMN Salary NOT NULL;  
GO
- D. ALTER TABLE Openings  
WITH NOCHECK  
ADD CONSTRAINT CT\_SALARY  
CHECK (Salary>=0);  
GO  
ALTER TABLE Openings  
WITH NOCHECK  
MODIFY COLUMN Salary NOT NULL;  
GO  
UPDATE Openings SET Salary=0 WHERE Salary IS NULL;  
GO

A. Option A B. Option B C. Option C  
D. Option D Answer: A Case Study 6: Coho Winery (Question 97 ~ Question 110) Overview You are a database developer for a company named Coho Winery. Coho Winery has an office in London. Coho Winery has an application that is used to process purchase orders from customers and retailers in 10 different countries. The application uses a web front end to process orders from the Internet. The web front end adds orders to a database named Sales. The Sales database is managed by a server named Server1. An empty copy of the Sales database is created on a server named Server2 in the London office. The database will store sales data for customers in Europe. A new version of the application is being developed. In the new version, orders will be placed either by using the existing web front end or by loading an XML file. Once a week, you receive two files that contain the purchase orders and the order details of orders from offshore facilities. You run the `usp_ImportOrders` stored procedure and the `usp_ImportOrderDetails` stored procedure to copy the offshore facility orders to the Sales database. The Sales database contains a table named Orders that has more than 20 million rows. Database Definitions Database and Tables The following scripts are used to create the database and its tables:

```
01 CREATE DATABASE Sales;
02 GO
03 USE Sales;
04 GO
05 CREATE TABLE Products
06 (
07     ProductID int IDENTITY(1,1) NOT NULL,
08     Name nvarchar(100) NOT NULL,
09     UnitPrice decimal(18,2) NOT NULL,
10     Discontinued bit NOT NULL DEFAULT 0,
11     CONSTRAINT PK_Products PRIMARY KEY (ProductID)
12 );
13 GO
14
15 CREATE TABLE Customers
16 (
17     CustomerID int IDENTITY(1,1) NOT NULL,
18     Name nvarchar(200) NOT NULL,
19     Email nvarchar(200) NOT NULL,
20     Phone nvarchar(10) NOT NULL,
21     Address1 nvarchar(200) NOT NULL,
22     Address2 nvarchar(200) NULL,
23     City nvarchar(200) NOT NULL,
24     State char(2) NOT NULL,
25     ZIP char(5) NOT NULL,
26     CONSTRAINT PK_Customers PRIMARY KEY (CustomerID)
27 );
28 GO
29
30 CREATE TABLE Orders
31 (
32     OrderID int IDENTITY(1,1) NOT NULL,
33     CustomerID int NOT NULL,
34     OrderDate datetime NOT NULL DEFAULT GETDATE(),
35     DeliveryDate datetime NOT NULL,
36     ShipDate datetime NULL,
37     Amount decimal(18,2) NOT NULL,
38     CONSTRAINT PK_Orders PRIMARY KEY (OrderID)
39 );
40 GO
41
42 ALTER TABLE Orders
43     ADD CONSTRAINT FK_Orders_Customers
44     FOREIGN KEY (CustomerID)
45     REFERENCES Customers (CustomerID);
46 GO
47
48 CREATE TABLE OrderDetails
49 (
50     OrderID int NOT NULL,
51     LineItem int NOT NULL,
52     ProductID int NOT NULL,
53     Quantity int NOT NULL,
54     UnitPrice decimal(18,2) NOT NULL,
55     Total decimal(18,2) NOT NULL,
56     Discount decimal(18,2) NULL,
57     CONSTRAINT PK_OrderDetails PRIMARY KEY (OrderID,
58 );
59 GO
60
61 ALTER TABLE OrderDetails
62     ADD CONSTRAINT FK_OrderDetails_Orders
63     FOREIGN KEY (OrderID)
64     REFERENCES Orders (OrderID);
65 GO
66
67 ALTER TABLE OrderDetails
68     ADD CONSTRAINT FK_OrderDetails_Products
69     FOREIGN KEY (ProductID)
70     REFERENCES Products (ProductID);
71 GO
```

Stored Procedures The following are the definitions of the stored procedures used in the database:

```
51 AS
52 SELECT OrderID, DeliveryDate, Amount
53 FROM Orders
54 WHERE ShipDate IS NULL
55 ORDER BY DeliveryDate;
56 GO
57
58 CREATE PROCEDURE usp_GetOrdersByProduct
59 @productID int
60
61 AS
62 SELECT OrderID, LineItem, Quantity,
63 UnitPrice, Total, Discount
64 FROM OrderDetails
65
66 WHERE ProductID = @productID;
67 GO
68
69 CREATE PROCEDURE usp_ImportOrders
70 AS
71 BULK INSERT Orders
72 FROM 'f:\orders\orders.tbl'
73 WITH
74 (
75 FIELDTERMINATOR = '|',
76 ROWTERMINATOR = '\n'
77 );
78 GO
79 CREATE PROCEDURE usp_ImportOrderDetails
80 @firstRow int
81 AS
82 BULK INSERT OrderDetails
83 FROM 'f:\orders\details.tbl'
84 WITH
85 (
86 FIRSTROW = @firstRow,
87 FIELDTERMINATOR = '|',
88 ROWTERMINATOR = '\n'
89 );
90 GO
91 GO
```

Indexes The following indexes are part of the Sales database:

```
01 CREATE INDEX IX_Orders_ShipDate
02 ON Orders (ShipDate)
03
04 INCLUDE (CustomerID, OrderDate, Amount);
05 GO
```

**Data Import** The XML files will contain the list of items in each order. Each retailer will have its own XML schema and will be able to use different types of encoding. Each XML schema will use a default namespace. The default namespaces are not guaranteed to be unique. For testing purposes, you receive an XSD file from a customer. For testing purposes, you also create an XML schema collection named ValidateOrder. ValidateOrder contains schemas for all of the retailers. The new version of the application must validate the XML file, parse the data, and store the parsed data along with the original XML file in the database. The original XML file must be stored without losing any data.

**Reported Issues**

**Performance Issues** You notice the following for the usp\_GetOrdersAndItems stored procedure: The stored procedure takes a long time to complete. Less than two percent of the rows in the Orders table are retrieved by usp\_GetOrdersAndItems. A full table scan runs when the stored procedure executes. The amount of disk space used and the amount of time required to insert data are very high. You notice that the usp\_GetOrdersByProduct stored procedure uses a table scan when the stored procedure is executed.

**Page Split Issues** Updates to the Orders table cause excessive page splits on the IX\_Orders\_ShipDate index.

**Requirements**

**Site Requirements** Users located in North America must be able to view sales data for customers in North America and Europe in a single report. The solution must minimize the amount of traffic over the WAN link between the offices.

**Bulk Insert Requirements** The usp\_ImportOrderDetails stored procedure takes more than 10 minutes to complete. The stored procedure runs daily. If the stored procedure fails, you must ensure that the stored procedure restarts from the last successful set of rows.

**Index Monitoring Requirements** The usage of indexes in the Sales database must be monitored continuously. Monitored data must be maintained if a server restarts. The monitoring solution must minimize the usage of memory resources and processing resources.

**QUESTION 97** You need to implement a solution that meets the site requirements. What should you implement? A. A non-indexed view on Server2 B. A distributed view on Server2 C. A distributed view on Server1

D. A non-indexed view on Server1 Answer: C QUESTION 98 You need to ensure that usp\_AddXMLOrder can be used to validate the XML input from the retailers. Which parameters should you add to usp\_AddXMLOrder on line 04 and line 05? (Each correct answer presents part of the solution. Choose all that apply.)

- A. @schema varbinary(100).
- B. @items varchar(max).
- C. @schema sysname.
- D. @items varbinary(max).
- E. @items xml.

F. @schema xml. Answer: CE QUESTION 99 You need to implement a solution that addresses the performance issues of the usp\_GetOrdersByProduct stored procedure. Which statement should you execute?

- A. CREATE INDEX IX\_Order ON OrderDetails (ProductID) INCLUDE (LineItem, Quantity)
- B. CREATE INDEX IX\_Order ON OrderDetails (ProductID) INCLUDE (OrderID, LineItem) [www.pass4it.com](http://www.pass4it.com)
- C. CREATE INDEX IX\_Order ON OrderDetails (ProductID) INCLUDE (OrderID, LineItem, Quantity)
- D. CREATE INDEX IX\_Order ON OrderDetails (ProductID) INCLUDE (LineItem, Quantity)

A. Option A B. Option B C. Option C

D. Option D Answer: C QUESTION 100 You plan to create a stored procedure that inserts data from an XML file to the OrderDetails table. The following is the signature of the stored procedure:

```
CREATE PROCEDURE usp_InsertItems  
@items XML (ValidateOrder)
```

T  
h  
e  
f  
ol  
lo  
w  
in  
g  
is  
th  
e  
X  
S  
D  
fi  
le  
u  
s  
e  
d  
to  
cr  
e  
at  
e



E

xp

or

t

da

te:

W

ed

M

ar

12

20

:3

5:

43

20

25

/

+

00

00

G

M

T

th

e

V

al

id

at

e

O

r

d

er

s

c

h

e

m

a

c

ol

le

ct

io

n:

```
<?xml version="1.0" encoding="UTF-16"?>
<xsd:schema
  xmlns:xsd="http://www.w3.org/2001/XMLSchema" >
<xsd:element name="root">
  <xsd:complexType mixed="true">
    <xsd:sequence>
      <xsd:element name="Product"
        minOccurs="1" maxOccurs="unbounded">
        <xsd:complexType mixed="true">
          <xsd:sequence>
            <xsd:element name="UnitPrice" type="xsd:decimal"
              minOccurs="1" maxOccurs="1" />
            <xsd:element name="Quantity" type="xsd:integer"
              minOccurs="1" maxOccurs="1" />
          </xsd:sequence>
          <xsd:attribute name="lineItem"
            type="xsd:integer" use="required"/>
          <xsd:attribute name="productID"
            type="xsd:integer" use="required"/>
        </xsd:complexType>
      </xsd:sequence>
      <xsd:attribute name="numberItems"
        type="xsd:integer" use="required"/>
    </xsd:complexType>
  </xsd:element>
</xsd:schema>
```

You develop a code segment that retrieves the number of items and loops through each item. Each time the loop runs, a variable named @itemNumber is incremented. You need to develop a code segment that retrieves the product ID of each item number in the loop. Which code segment should you develop? A. SET @productID = @items.value('/Root/Product/productID', int) B. SET @productID = @items.value('/Root/Product[' + @itemNumber + ']/@productID', int) C. SET @productID = @items.value('/Root/Product/@productID', int) D. SET @productID = @items.value('/Root/Product[' + @itemNumber + ']/productID', int) Answer: D



<http://www.passleader.com/70-464.html> QUESTION 101 You need to ensure that a new execution plan is used by usp\_GetOrdersByProduct each time the stored procedure runs. What should you do? A. Add WITH (FORCESEEK) to line 07 in usp\_GetOrdersByProduct. B. Execute sp\_recompile 'usp\_GetOrdersByProduct'. C. Execute sp\_help 'usp\_GetOrdersByProduct'. D. Add WITH RECOMPILE to line 03 in usp\_GetOrdersByProduct. Answer: D QUESTION 102 You need to implement a solution that addresses the index monitoring requirements. What should you do? A. Schedule a SQL Server Agent job that saves data from the dynamic management views to a table in the database. B. Create a SQL Server Audit that saves data to a log file, and then create a SQL Server Audit Specification that gathers data from the DATABASE\_OPERATION group. C. Create a performance monitor Data Collector Set (DCS) that monitors the SQL Server counters. D. Schedule a SQL Server Profiler trace, and then save the trace data to a table in the database. Answer: A QUESTION 103 You need to implement a solution that addresses the page split issues. Which statement should you execute? A. ALTER INDEX IX\_Orders\_ShipDate ON Orders REBUILD WITH (PAD\_INDEX = OFF, DROP\_EXISTING = ON); B. ALTER INDEX IX\_Orders\_ShipDate ON Orders REBUILD WITH (FILLFACTOR=50, DROP\_EXISTING = ON); C. ALTER INDEX IX\_Orders\_ShipDate ON Orders REBUILD WITH (FILLFACTOR=0, DROP\_EXISTING = ON); D. ALTER INDEX IX\_Orders\_ShipDate ON Orders REBUILD WITH (PAD\_INDEX=ON/ DROP\_EXISTING = ON); Answer: B QUESTION 104 You need to implement a solution that solves the performance issues of usp\_GetOrdersAndItems. Which statements should you execute? A. CREATE INDEX IX\_Orders\_Active ON Orders(ShipDate, DeliveryDate, Amount) B. CREATE INDEX IX\_Orders\_Active ON Orders(DeliveryDate) INCLUDE(Amount) WHERE ShipDate



IS NXXL C.&#160;&#160;&#160; CREATE INDEX IX\_Orders\_Active ON Orders(DeliveryDate, Amount) WHERE ShipDate IS NULL D.&#160;&#160;&#160; CREATE INDEX IX\_Orders\_Active ON Orders(ShipDate, DeliveryDate) INCLUDE(Amount) Answer: B QUESTION 105 You need to modify usp\_GetOrdersAndItems to ensure that an order is NOT retrieved by usp\_GetOrdersAndItems while the order is being updated. What should you add to usp\_GetOrdersAndItems?

A.&#160;&#160;&#160; Add (READPAST) to the end of line 06. B.&#160;&#160;&#160; Add SET TRANSACTION ISOLATION LEVEL SNAPSHOT to line 03. C.&#160;&#160;&#160; Add SET TRANSACTION ISOLATION LEVEL SERIALIZABLE to line 03. D.&#160;&#160;&#160; Add (UPDLOCK) to the end of line 06. Answer: A QUESTION 106 You need to implement a solution that addresses the bulk insert requirements. What should you add to line 08 in usp\_ImportOrderDetails? A.&#160;&#160;&#160; LASTROW=0. B.&#160;&#160;&#160; BATCHSIZE=0. C.&#160;&#160;&#160; BATCHSIZE=1000. D.&#160;&#160;&#160; LASTROW = 1000. Answer: C QUESTION 107 You discover that the usp\_GetOrdersAndItems stored procedure takes a long time to complete while usp\_AddOrder or usp\_AddXMLOrder run. You need to ensure that usp\_GetOrdersAndItems completes as quickly as possible. What should you do? (Each correct answer presents part of the solution. Choose all that apply.) A.&#160;&#160;&#160; Set the isolation level of the usp\_GetOrdersAndItems stored procedure to SERIALIZABLE. B.&#160;&#160;&#160; Execute the ALTER DATABASE Sales SET ALLOW\_SNAPSHOT\_ISOLATION ON statement. C.&#160;&#160;&#160; Set the isolation level of the usp\_AddOrder stored procedure to SERIALIZABLE. D.&#160;&#160;&#160; Set the isolation level of the usp\_GetOrdersAndItems stored procedure to SNAPSHOT. E.&#160;&#160;&#160; Set the isolation level of the usp\_AddOrder stored procedure to SNAPSHOT. F.&#160;&#160;&#160; Execute the ALTER DATABASE Sales SET ALLOWSNAPSHOTISOLATION OFF statement.

Answer: BD QUESTION 108 You need to modify the Orders table to store the XML data used by the retailers. Which statement should you execute? A.&#160;&#160;&#160; ALTER Orders ADD originalOrder XML (ValidateOrder); B.&#160;&#160;&#160; ALTER Orders ADD originalOrder XML; C.&#160;&#160;&#160; ALTER Orders ADD originalOrdervarchar(max); D.&#160;&#160;&#160; ALTER Orders ADD originalOrdervarbinary(max); Answer: D QUESTION 109 You need to modify usp.GetOrdersAndItems to ensure that an order is NOT retrieved by usp\_GetOrdersAndItems while the order is being updated. What should you add to usp.GetOrdersAndItems? A.&#160;&#160;&#160; Add WITH (NOLOCK) to the end of line 47. B.&#160;&#160;&#160; Add SET TRANSACTION ISOLATION LEVEL READ COMMITTED to line 44. C.&#160;&#160;&#160; Add SET TRANSACTION ISOLATION LEVEL READ UNCOMMITTED to line 44. D.&#160;&#160;&#160; Add WITH (READPAST) to the end of line 47. Answer: B QUESTION 110 You need to ensure that a new execution plan is used by usp\_GetOrdersByProduct each time the stored procedure runs. What should you do? A.&#160;&#160;&#160; Execute sp\_help usp\_GetOrdersByProduct. B.&#160;&#160;&#160; Add WITH (FORCESEEK) to line 69 in usp.GetOrdersByProduct. C.&#160;&#160;&#160; Add WITH RECOMPILE to line 64 in usp.GetOrdersByProduct. D.&#160;&#160;&#160; Execute sp\_recompile usp.GetOrdersByProduct'. Answer: B

Compare And Choose The Best **PassLeader 70**

	
<p>Banned By Microsoft Not Available</p>	<p>Leader of IT Certification 153 Q&amp;As Price: \$99.99 Coupon Code -- CELEB</p>

<http://www.passleader.com/70-464.html>