

## [New Exam Dumps Practice PassLeader New 168q 70-461 Dumps To Pass Exam Quickly

New Updated 70-461 Exam Questions from PassLeader 70-461 PDF dumps! Welcome to download the newest PassLeader 70-461 VCE dumps: <http://www.passleader.com/70-461.html> (168 Q&As) Keywords: 70-461 exam dumps, 70-461 exam questions, 70-461 VCE dumps, 70-461 PDF dumps, 70-461 practice tests, 70-461 study guide, 70-461 braindumps, Querying Microsoft SQL Server 2012 Exam **NEW QUESTION 158** You have a Microsoft SQL Server database that includes two tables named EmployeeBonus and BonusParameters. The tables are defined by using the following Transact-SQL statements:

```
CREATE TABLE [EmployeeBonus] ([EmpNumber] [int] NOT NULL, [Quarterly] [tinyint] NOT NULL, [HalfYearly] [tinyint] NOT NULL, [Yearly] [tinyint] NOT NULL) ON [PRIMARY]
```

```
CREATE TABLE [BonusParameters] ([AvailableBonus] [float] NOT NULL, [CompanyPerformance] [float] NOT NULL) ON [PRIMARY]
```

The tables are used to compute a bonus for each employee. The EmployeeBonus table has a non- null value in either the Quarterly, HalfYearly or Yearly column. This value indicates which type of bonus an employee receives. The BonusParameters table contains one row for each calendar year that stores the amount of bonus money available and a company performance indicator for that year. You need to calculate a bonus for each employee at the end of a calendar year. Which Transact-SQL statement should you use? A. `SELECT CAST(CHOOSE((Quarterly * AvailableBonus * CompanyPerformance)/40, (HalfYearly * AvailableBonus * CompanyPerformance)/20, (Yearly * AvailableBonus * CompanyPerformance)/10) AS money) AS `Bonus` FROM EmployeeBonus, BonusParameters` B. `SELECT "Bonus" = CASE EmployeeBonus WHEN Quarterly=1 THEN (Quarterly * AvailableBonus * CompanyPerformance)/40 WHEN HalfYearly=1 THEN (HalfYearly * AvailableBonus * CompanyPerformance)/20 WHEN Yearly=1 THEN (Yearly * AvailableBonus * CompanyPerformance)/10 END FROM EmployeeBonus, BonusParameters` C. `SELECT CAST(COALESCE((Quarterly * AvailableBonus * CompanyPerformance)/40, (HalfYearly * AvailableBonus * CompanyPerformance)/20, (Yearly * AvailableBonus * CompanyPerformance)/10) AS money) AS `Bonus` FROM EmployeeBonus, BonusParameters` D. `SELECT NULLIF(NULLIF((Quarterly * AvailableBonus * CompanyPerformance)/40, (HalfYearly * AvailableBonus * CompanyPerformance)/20), (Yearly * AvailableBonus * CompanyPerformance)/10) AS `Bonus` FROM EmployeeBonus, BonusParameters` **Answer: B** **NEW QUESTION 159** You use Microsoft SQL Server 2012 to develop a database application. You need to create an object that meets the following requirements: - Takes an input parameter - Returns a table of values - Can be referenced within a view Which object should you use? A. inline table-valued function B. user-defined data type C. stored procedure D. scalar-valued function **Answer: A** **Explanation:** Incorrect answers: Not B: A user-defined data type would not be able to take an input parameter. Not C: A stored procedure cannot be used within a view. Not D: A scalar-valued would only be able to return a single simple value, not a table. **NEW QUESTION 160** .....

Download the newest PassLeader 70-461 dumps from [passleader.com](http://www.passleader.com) now! 100% Pass Guarantee! ? <http://www.passleader.com/70-461.html> **NEW QUESTION 162** You are developing a Microsoft SQL Server 2012 database for a company. The database contains a table that is defined by the following Transact-SQL statement:

```
CREATE TABLE [dbo].[Employees] ([EmpNumber] [int] NOT NULL, [Surname] [varchar](40) NOT NULL, [GivenName] [varchar](20) NOT NULL, [PersonalIDNumber] [varchar](11) NOT NULL, [Gender] [varchar](1) NULL, [DateOfBirth] [date] NOT NULL)
```

You use the following Transact-SQL script to insert new employee data into the table. Line numbers are included for reference only.

```
01 BEGIN TRY
02 INSERT INTO [dbo].[Employees]([EmpNumber],[Surname],[GivenName],[Gender],[DateOfBirth],[PersonalIDNumber])
03 VALUES (132,'Williams','John', 'M', '1/1/1990',NULL)
04 END TRY
05 BEGIN CATCH
06
07 END CATCH
```

[www.passleader.com](http://www.passleader.com)

If an error occurs, you must report the error message and line number at which the error occurred and continue processing errors. You need to complete the Transact-SQL script. Which Transact-SQL segment should you insert at line 06?

A. &#160;&#160;&#160; SELECT ERROR\_LINE(), ERROR\_MESSAGE() B. &#160;&#160;&#160; DECLARE @message NVARCHAR(1000),@severity INT, @state INT; SELECT @message = ERROR\_MESSAGE(), @severity = ERROR\_SEVERITY(), @state = ERROR\_STATE(); RAISERROR (@message, @severity, @state); C. &#160;&#160;&#160; DECLARE @message NVARCHAR(1000),@severity INT, @state INT; SELECT @message = ERROR\_MESSAGE(), @severity = ERROR\_SEVERITY(), @state = ERROR\_STATE(); THROW (@message, @severity, @state); D. &#160;&#160;&#160; THROW; **Answer: B Explanation:** When the code in the CATCH block finishes, control passes to the statement immediately after the END CATCH statement. Errors trapped by a CATCH block are not returned to the calling application. If any part of the error information must be returned to the application, the code in the CATCH block must do so by using mechanisms such as SELECT result sets or the RAISERROR and PRINT statements. Reference: TRY...CATCH (Transact-SQL)

<https://msdn.microsoft.com/en-us/library/ms175976.aspx> **NEW QUESTION 164** You are maintaining a Microsoft SQL Server database. You run the following query:

```
SELECT
    e.[ID]
    p.[Title],
    p.[GivenName] + ' ' + p.[SurName],
    e.[JobTitle],
    edh.[StartDate]
FROM [ActiveEmployee] e
INNER JOIN [Person] p ON p.[ID] = e.[ID]
INNER JOIN [History] edh ON e.[ID] = edh.[ID]
WHERE edh.EndDate IS NULL
```

[www.passleader.com](http://www.passleader.com)

You observe performance issues when you run the query. You capture the following query execution plan:



You need to ensure that the query performs returns the results as quickly as possible. Which action should you perform?  
A. &#160;&#160;&#160; Add a new index to the ID column of the Person table. B. &#160;&#160;&#160; Add a new index to the EndDate column of the History table. C. &#160;&#160;&#160; Create a materialized view that is based on joining data from the ActiveEmployee and History tables. D. &#160;&#160;&#160; Create a computed column that concatenates the GivenName and SurName columns. **Answer: A Explanation:** Cost is 53% for the Table Scan on the Person (p) table. This table scan is on the ID column, so we should put an index on it. **NEW QUESTION 165** ..... Download the newest PassLeader 70-461 dumps from [passleader.com](http://www.passleader.com) now! 100% Pass Guarantee! ? <http://www.passleader.com/70-461.html> **NEW QUESTION 167** You are developing a database in SQL Server 2012 to store information about current employee project assignments. You are creating a view that uses data from the project assignment table. You need to ensure that the view does not become invalid if the schema of the project assignment table changes. What should you do? A. &#160;&#160;&#160; Create the view by using an account in the sysadmin role. B. &#160;&#160;&#160; Add a DDL trigger to the project assignment table to re-create the view after any schema change. C. &#160;&#160;&#160; Create the view in a new schema. D. &#160;&#160;&#160; Add a DDL trigger to the view to

block any changes. **Answer: B Explanation:** DDL triggers are a special kind of trigger that fire in response to Data Definition Language (DDL) statements. They can be used to perform administrative tasks in the database such as auditing and regulating database operations. Reference: DDL Triggers [https://technet.microsoft.com/en-us/library/ms190989\(v=sql.105\).aspx](https://technet.microsoft.com/en-us/library/ms190989(v=sql.105).aspx) **NEW**

**QUESTION 168** You are maintaining a Microsoft SQL Server database that stores order information for an online store website.

The database contains a table that is defined by the following Transact-SQL statement:

```
CREATE TABLE [dbo].[SalesOrderHeader] (  
    [SalesOrderID] [int] IDENTITY(1,1) NO  
    [OrderDate] [datetime] NOT NULL,  
    [Status] [tinyint] NOT NULL,  
    [PurchaseOrderNumber] [nvarchar](25)  
    [AccountNumber] [nvarchar](15) NULL,  
    [CustomerID] [int] NOT NULL,  
    [TotalDue] [money] NOT NULL,  
    CONSTRAINT [PK_SalesOrderHeader] PRIMAR  
    (  
        [SalesOrderID] ASC  
    ) ON [PRIMARY]
```

[www.pdf](http://www.pdf)

You need to ensure that purchase order numbers are used only for a single order. What should you do? A. Create a new CLUSTERED constraint on the PurchaseOrderNumber column. B. Create a new UNIQUE constraint on the PurchaseOrderNumber column. C. Create a new PRIMARY constraint on the PurchaseOrderNumber column. D. Create a new FOREIGN KEY constraint on the PurchaseOrderNumber column. **Answer: B Explanation:** You can use UNIQUE constraints to make sure that no duplicate values are entered in specific columns that do not participate in a primary key. Although both a UNIQUE constraint and a PRIMARY KEY constraint enforce uniqueness, use a UNIQUE constraint instead of a PRIMARY KEY constraint when you want to enforce the uniqueness of a column, or combination of columns, that is not the primary key. Reference: UNIQUE Constraints [https://technet.microsoft.com/en-us/library/ms191166\(v=sql.105\).aspx](https://technet.microsoft.com/en-us/library/ms191166(v=sql.105).aspx) Download the newest PassLeader 70-461 dumps from passleader.com now! 100% Pass Guarantee! 70-461 PDF dumps & 70-461 VCE dumps: <http://www.passleader.com/70-461.html> (168 Q&As)