EDUSUM

#1 Online Certification Guide

70-483

MCSA Universal Windows Platform

A Success Guide to Prepare-
Programming in C#

edusum.com

Table of Contents

Introduction to 70-483 Exam on Programming in C#

Use this quick start guide to collect all the information about Microsoft Programming in C# (70-483) Certification exam. This study guide provides a list of objectives and resources that will help you prepare for items on the 70-483 Programming in C# exam. The Sample Questions will help you identify the type and difficulty level of the questions and the Practice Exams will make you familiar with the format and environment of an exam. You should refer this guide carefully before attempting your actual Microsoft MCSA Universal Windows Platform certification exam.

The Microsoft Programming in C# certification is mainly targeted to those candidates who want to build their career in Microsoft Visual Studio domain. The Microsoft Certified Solutions Associate (MCSA) - Universal Windows Platform exam verifies that the candidate possesses the fundamental knowledge and proven skills in the area of Microsoft MCSA Universal Windows Platform.

Microsoft 70-483 Certification Details:

Exam Name	Microsoft Certified Solutions Associate (MCSA) - Universal Windows Platform
Exam Code	70-483
Exam Price	$165 (USD)
Duration	120 min
Number of Questions	45-55
Passing Score	700 / 1000
Books / Training	20483B
Schedule Exam	Pearson VUE
Sample Questions	Microsoft Programming in C# Sample Questions
Practice Exam	**Microsoft 70-483 Certification Practice Exam**

Microsoft 70-483 Exam Syllabus:

Topic	Details	Weights
Manage program flow	Implement multithreading and asynchronous processing - Use the Task Parallel library (ParallelFor, Plinq, Tasks); create continuation tasks; spawn threads by using ThreadPool; unblock the UI; use async and await keywords; manage data by using concurrent collections Manage multithreading - Synchronize resources; implement locking; cancel a long-running task; implement thread-safe methods to handle race conditions Implement program flow - Iterate across collection and array items; program decisions by using switch statements, if/then, and operators; evaluate expressions Create and implement events and callbacks - Create event handlers; subscribe to and unsubscribe from events; use built-in delegate types to create events; create delegates; lambda expressions; anonymous methods Implement exception handling - Handle exception types (SQL exceptions, network exceptions, communication exceptions, network timeout exceptions); catch typed vs. base exceptions; implement try-catch-finally blocks; throw exceptions; determine when to rethrow vs. throw; create custom exceptions	25-30%
Create and use types	Create types - Create value types (structs, enum), reference types, generic types, constructors, static variables, methods, classes, extension methods, optional and named parameters, and indexed properties; create overloaded and overriden methods Consume types - Box or unbox to convert between value types; cast types; convert types; handle dynamic types; ensure interoperability with unmanaged code, for example, dynamic keyword	25-30%

Topic	Details	Weights
	Enforce encapsulation - Enforce encapsulation by using properties, by using accessors (public, private, protected), and by using explicit interface implementation Create and implement a class hierarchy - Design and implement an interface; inherit from a base class; create and implement classes based on the IComparable, IEnumerable, IDisposable, and IUnknown interfaces Find, execute, and create types at runtime by using reflection - Create and apply attributes; read attributes; generate code at runtime by using CodeDom and lambda expressions; use types from the System.Reflection namespace (Assembly, PropertyInfo, MethodInfo, Type) Manage the object life cycle - Manage unmanaged resources; implement IDisposable, including interaction with finalization; manage IDisposable by using the Using statement; manage finalization and garbage collection Manipulate strings - Manipulate strings by using the StringBuilder, StringWriter, and StringReader classes; search strings; enumerate string methods; format strings	
Debug applications and implement security	Validate application input - Validate JSON data; data collection types; manage data integrity; evaluate a regular expression to validate the input format; use built-in functions to validate data type and content out of scope: writing regular expressions Perform symmetric and asymmetric encryption - Choose an appropriate encryption algorithm; manage and create certificates; implement key management; implement the System.Security namespace; hashing data; encrypt streams Manage assemblies - Version assemblies; sign assemblies using strong names; implement side-by-side hosting; put an assembly in the global assembly cache; create a WinMD assembly	25-30%

Topic	Details	Weights
	Debug an application - Create and manage compiler directives; choose an appropriate build type; manage programming database files and symbols Implement diagnostics in an application - Implement logging and tracing; profiling applications; create and monitor performance counters; write to the event log	
Implement data access	Perform I/O operations - Read and write files and streams; read and write from the network by using classes in the System.Net namespace; implement asynchronous I/O operations Consume data - Retrieve data from a database; update data in a database; consume JSON and XML data; retrieve data by using web services Query and manipulate data and objects by using LINQ - Query data by using operators (projection, join, group, take, skip, aggregate); create method-based LINQ queries; query data by using query comprehension syntax; select data by using anonymous types; force execution of a query; read, filter, create, and modify data structures by using LINQ to XML Serialize and deserialize data - Serialize and deserialize data by using binary serialization, custom serialization, XML Serializer, JSON Serializer, and Data Contract Serializer Store data in and retrieve data from collections - Store and retrieve data by using dictionaries, arrays, lists, sets, and queues; choose a collection type; initialize a collection; add and remove items from a collection; use typed vs. non-typed collections; implement custom collections; implement collection interfaces	25-30%

70-483 Sample Questions:

01. When you create an abstract method, how do you use that method in a derived class?
a) You must overload the method in your derived class.
b) You must override the method in your derived class.
c) Abstract methods cannot be used in derived classes.
d) You need to declare the method as virtual in your derived class.

02. Which the following statements about the this keyword is false?
a) A constructor can use at most one this statement.
b) A constructor can use a this statement and a base statement if the base statement comes first.
c) The this keyword lets a constructor invoke a different constructor in the same class.
d) If a constructor uses a this statement, its code is executed after the invoked constructor is executed.

03. How do you execute a method as a task?
a) Create a new Task object, and then call the Start method on the newly created object.
b) Create the task via the Task.Run method.
c) Create the task via the Task.Factory.StartNew method.
d) All the above.
e) None of the above.

04. What is a strong name assembly?
a) An assembly with the name marked as bold
b) An assembly with a major and minor version specified
c) An assembly with a full version specified
d) An assembly with the culture info specified
e) A signed assembly

05. Which statement selects an anonymous type?
a) select { h.City, h.State }
b) select h
c) select new { h.City, h.State }
d) select h.City, h.State

06. Which of the following is a valid delegate definition?
a) private delegate float MyDelegate(float);
b) private delegate MyDelegate(x);
c) private delegate MyDelegate(float x);
d) private delegate void MyDelegate(float x);

07. Which properties of an ADO.NET Command object must you set to execute a stored procedure?
a) CommandTypeStoredProcedureNameParameters
b) IsStoredProcedureCommandTypeStoredProcedureNameParameters
c) CommandTypeCommandTextParameters
d) IsStoredProcedureCommandTextParameters

08. How can you deploy a private assembly?
a) By running gacutil.exe
b) By adding a reference to the assembly in Visual Studio
c) By copying the file in the Bin folder of the application
d) By copying the file in C:\Windows folder

09. Which of the following statements about statement lambdas is false?
a) A statement lambda can include more than one statement.
b) A statement lambda cannot return a value.
c) A statement lambda must use braces, { }.
d) If a statement lambda returns a value, it must use a return statement.

10. Which where clause returns all integers between 10 and 20?
a) where i >= 10 and i <= 20
b) where i >= 10 && i <= 20
c) where i gt 10 and i lt 20
d) where i gt 10 && i lt 20

Answers to 70-483 Exam Questions:

Question: 01 Answer: b	Question: 02 Answer: b	Question: 03 Answer: d	Question: 04 Answer: e	Question: 05 Answer: c
Question: 06 Answer: d	Question: 07 Answer: c	Question: 08 Answer: b, c	Question: 09 Answer: b	Question: 10 Answer: b

Note: If you find any typo or data entry error in these sample questions, we request you to update us by commenting on this page or write an email on feedback@edusum.com