Quiz review

Started on	Tuesday, 14 March 2023, 10:38 PM
State	Finished
Completed on	Tuesday, 14 March 2023, 10:46 PM
Time taken	8 mins 15 secs
Marks	8/10
Grade	**80** out of 100

Question **1**

Correct

Mark 1 out of 1

595

Which of the following statement is true?

Select one:

○ You can call methods through variables of delegate types.

○ A delegate is an object that holds a reference to a method.

○ Via delegates, you can assign methods to variables, and pass methods to and from other methods.

◉ All of these statements are true. ✔

Your answer is correct.

595

The correct answer is: All of these statements are true.

Question **2**

Correct

Mark 1 out of 1

What is the purpose of using a delegate in C#?

Select one:

○ a. To create a pointer to a method   595

◉ b. To encapsulate a method as an object ✔

○ c. To implement polymorphism

○ d. To define a new data type

Your answer is correct.

The correct answer is: To encapsulate a method as an object

Correct

Mark 1 out of 1

What is the syntax for declaring a delegate in C#?

Select one:

- ○ a.   function return_type delegate_name(parameters);
- ◉ b.   delegate return_type delegate_name(parameters); ✔
- ○ c.   delegate_name(parameters) => return_type;
- ○ d.   event return_type delegate_name(parameters);

Your answer is correct.

The correct answer is: delegate return_type delegate_name(parameters);

595

Question **4**

Incorrect

Mark 0 out of 1

What is a delegate in C#?

Select one:

- ○ a.   A type that defines a method signature
- ○ b.   A class that can be inherited from 595
- ○ c.   A keyword used to define a variable
- ◉ d.   An object that can reference a method ✖

Your answer is incorrect.

The correct answer is: A type that defines a method signature

Question **5**

Incorrect

595

Mark 0 out of 1

What is the difference between a delegate and an event in C#?

Select one:

- ○ a.   An event cannot be invoked directly, while a delegate can
- ○ b.   A delegate can only reference one method, while an event can reference multiple methods
- ◉ c.   An event is a special type of delegate ✖
- ○ d.   Events only work with user-defined classes, while delegates can work with any type

---

Your answer is incorrect.

The correct answer is: A delegate can only reference one method, while an event can reference multiple methods

∧

Correct

Mark 1 out of 1

A delegate when wraps more than one method is said to be _____

Select one:

○ Event

◉ Multicast Delegate✔

○ Multi Delegate

○ Multidimensional Delegate

Your answer is correct.

The correct answer is: Multicast Delegate

595

Question **7**

Correct

Mark 1 out of 1

When would you use a lambda expression instead of a delegate in C#?

Select one:

○ a. When the delegate is used to create an event.

○ b. When the logic being defined is complex and requires multiple statements.

◉ c. When the logic being defined is short and simple.✔

○ d. When the delegate is defined within a class.

Your answer is correct.

The correct answer is: When the logic being defined is short and simple.

Question **8**

Correct

595

Mark 1 out of 1

How do you create a delegate in C#?

Select one:

◉ a. Using the "delegate" keyword followed by the method signature✔

○ b. By inheriting from the Delegate class

○ c. By using the "function" keyword

○ d. By instantiating a new delegate object

Your answer is correct.

The correct answer is: Using the "delegate" keyword followed by the method signature

∧

Correct

Mark 1 out of 1

What is the syntax for declaring a lambda expression in C#?

Select one:

○ a. (parameters) -> expression or statement;

◉ b. (parameters) => {expression or statement}; ✔

○ c. lambda(parameters) {expression or statement};

○ d. function(parameters) => expression or statement;

Your answer is correct.

The correct answer is: (parameters) => {expression or statement};

595

Question **10**

Correct

Mark 1 out of 1

Which of the following methods can be referred by the MyDelegate instance

delegate void MyDelegate(int x, int y);

Select one or more: 595

☑ static void MethodA(int a, int b) { } ✔

☐ static int MethodB(int a, int b) { return 0; }

☑ void MethodC(int a, int b) { } ✔

☐ void MethodD(float a, int b) { }

Your answer is correct.

The correct answers are: static void MethodA(int a, int b) { }, void MethodC(int a, int b) { }

595

Jump to...

∧