Quiz Details

Quiz Instructions:

☑ Show Question Details

Enumerations Quiz Pick 12 questions, 1 pts per question

Question

What keyword is used to declare an enumeration in C#?

- ○ enum
- ○ enumeration
- ○ Enum
- ○ Enumeration

Question

Which of these follows the convention for an enumeration name?

- ○ Gender
- ○ gender
- ○ _gender
- ○ GENDER

Question

How many bytes is an int?

- 4
- 2
- 1
- 8

Question

Which of the following shows the correct syntax to cast something to a double?

- (double)
- {double}
- double()
- double.Cast

Question

What value does **Neutered** have in the following enumeration?

enum Gender { Male = 2, Female, Spayed, Neutered, Unknown };

- 5
- 4

- 3
- 0

Question

What is the correct way to access the **Unknown** item of this enumeration?

enum Gender { Male, Female, Spayed, Neutered, Unknown };

- Gender.Unknown
- Unknown
- Gender{Unknown}
- Gender(Unknown)

Question

The default data type for an enumeration is integer.

- True
- False

Question

Which of the following is the correct syntax to use something other than the default data type in an enumeration?

- enum Gender : byte { Male, Female, Spayed, Neutered, Unknown };
- enum byte : Gender { Male, Female, Spayed, Neutered, Unknown };
- enum Gender { Male : byte, Female, Spayed, Neutered, Unknown };
- byte enum Gender { Male, Female, Spayed, Neutered, Unknown };

Question

Which of the following is an example of **implicit conversion**?

- double dblCost = 3;
- int intAge = 3.0;
- int intAge = (int) 3.0;
- double dblCost = (double) 3;

Question

Which of the following is NOT true about enumerations in C#?

- they make your code more difficult to read
- they make your code easier to read
- they make it easier to define a set of related integer constants
- they can help you dispose of "hard coded" values in your programs

Question

You can have decimal enums in C#.

- ○ True
- ○ False

Question

Casting is required in C-type languages when:

(There are two correct answers).

- ☐ information may be lost in a conversion
- ☐ converting from a derived class to a base class
- ☐ converting from a base class to a derived class
- ☐ converting between non-compatible types