1. What type would you choose for the following “numbers”?

* A person’s telephone number
  + Int/long/uint/ulong
* A person’s height
  + float
* A person’s age
  + short
* A person’s gender (Male, Female, Prefer Not To Answer)
  + string
* A person’s salary
  + decimal
* A book’s ISBN
  + Uint/ulong/int/long
* A book’s price
  + decimal
* A book’s shipping weight
  + Float/double
* A country’s population
  + Int/long/uint/ulong
* The number of stars in the universe
  + Long/ulong
* The number of employees in each of the small or medium businesses in theUnited Kingdom (up to about 50,000 employees per business)
  + Uint/long/ulong

2. What are the difference between value type and reference type variables? What is boxing and unboxing?

* value type will directly hold the value, while reference type will hold the memory address or reference for this value
* value type are stored in stack memory, while reference types will be stored in heap memory
* value type will not be collected by garbage collector, while reference type will be collected by garbage collector
* value type can be created by Struct or Enum, but reference type can be created by class, interface, delegate, or array
* value type cannot accept null values, but reference types can accept null values
* boxing: convert a value type to a reference type
* unboxing: convert the reference type to a value type

3. What is meant by the terms managed resource and unmanaged resource in .NET

* Managed resources are those that are pure .NET code and managed by the runtime and are under its direct control.
* Unmanaged resources are those that are not. File handles, pinned memory, COM objects, database connections etc.

4. What’s the purpose of Garbage Collector in .NET?

* The garbage collector manages the allocation and release of memory for an application. For developers working with managed code, this means that you don't have to write code to perform memory management tasks. Automatic memory management can eliminate common problems, such as forgetting to free an object and causing a memory leak or attempting to access memory for an object that's already been freed.

1. What happens when you divide an int variable by 0?
   1. The DivideByZeroException will be thrown when there is an attempt to divide an integral value by zero.
2. What happens when you divide a double variable by 0?
   1. It will return infinity or NaN.
3. What happens when you overflow an int variable, that is, set it to a value beyond its range?
   1. An OverflowException will be thrown
4. What is the difference between x = y++; and x = ++y;?
   1. The result of x++ is the value of x before the operation
   2. The result of ++x is the value of x after the operation
5. What is the difference between break, continue, and return when used inside a loop statement?
   1. The break statement terminates the closest enclosing iteration statement (that is, for, foreach, while, or do loop) or switch statement.
   2. The continue statement starts a new iteration of the closest enclosing iteration statement (that is, for, foreach, while, or do loop)
   3. The return statement terminates execution of the function in which it appears and returns control and the function's result, if any, to the caller.
6. What are the three parts of a for statement and which of them are required?
   1. The initializer section that is executed only once, before entering the loop. Typically, you declare and initialize a local loop variable in that section. The declared variable can't be accessed from outside the for statement.
   2. The condition section that determines if the next iteration in the loop should be executed. If it evaluates to true or is not present, the next iteration is executed; otherwise, the loop is exited. The condition section must be a Boolean expression.
   3. The iterator section that defines what happens after each execution of the body of the loop.
   4. All the sections of the for statement are optional.
7. What is the difference between the = and == operators?
   1. The assignment operator = assigns the value of its right-hand operand to a variable, a property, or an indexer element given by its left-hand operand. The result of an assignment expression is the value assigned to the left-hand operand. The type of the right-hand operand must be the same as the type of the left-hand operand or implicitly convertible to it.
   2. The equality operator == returns true if its operands are equal, false otherwise.
8. Does the following statement compile? for ( ; true; ) ;
   1. Yes, and it will be an infinite loop.
9. What does the underscore \_ represent in a switch expression?
   1. The underscore (\_) character replaces the default keyword to signify that it should match anything if reached.
10. What interface must an object implement to be enumerated over by using the foreach statement?
    1. The IEnumerable interface permits enumeration by using a foreach loop.