**C# Homework 02**

**Question 1**

What is a local variable?

**Answer**

They are variables that exist only in a method or another small section of code.

**Question 2**

What is a statement?

**Answer**

A statement is a command that performs and action, such as calculating a value and storing the result or displaying a message to a user. A statement is terminated by a semicolon.

**Question 3**

What is an identifier?

**Answer**

Identifiers are the names that you use to identify the elements in your programs, such as namespaces, classes, methods, and variables. Rules for identifiers you can only use letters, digits, underscores, case sensitive, and cannot begin with a number.

**Question 4**

What is keyword?

**Answer**

The C# language reservers 77 identifiers for its own use, and you cannot reuse these identifiers for your own purposes. These identifiers are called keywords, and each has a particular meaning. Examples of keywords are class, namespace, and using.

**Question 5**

What is a variable?

**Answer**

A variable is a storage location that holds a value. A variable is a named location in computer memory

**Question 6**

How do you declare a variable? How do you assign a value to a variable? Can you have a variable that does not have a value? Can you declare and initialize a variable in a single statement?

**Answer**

When you declare a variable you must specify the type of data it will hold. You declare the type and name of a variable in a declaration statement. Before you can use an unassigned variable you must assign it that is called the definite assignment rule. To assign a value to a variable you use an equal sign. Yes you can have a variable without a value. The process of assigning a value to a variable for the first time is initialize. Yes you can.

Int pi; (this variable has no value)

Double pi = 3.1415926;

E =2.718281828;

Double e

**Question 7**

What does Visual Studio when you open a project (a .csproj) without opening the solution (a .sln file)? Why might this be a problem?

**Answer**

Visual Studio will create its own solution file. It might be confusing for yourself because you it might not be what you need.

**Question 8**

How are arithmetic operators and variable types related?

**Answer**

If you have two numeric types you can multiply them together. If you have two nonnumeric types you can not multiply them together. Variables have certain attributes: name, memory location, type

Int a = 3;

Int b = 3;

Int c = a + b

Double cc = a / b;

Wht is cc equal to? // 0.75

What is c equal to? // 7

String d = “Gerge”;

String e = “ ”;

String f = “Washingon”;

String g = d + e + f;

What is g equal to? “George Washington”

What is d / f equal to?

ZIP\_CODES

Int Anthony\_zip = 22401;

Int ray\_zip = 22554;

Anthony\_zip + ray\_zip

**Question 9**

How do you turn an integer into a string?

**Answer**

Int a = 5;

String b = a.ToString(); // b = “5”

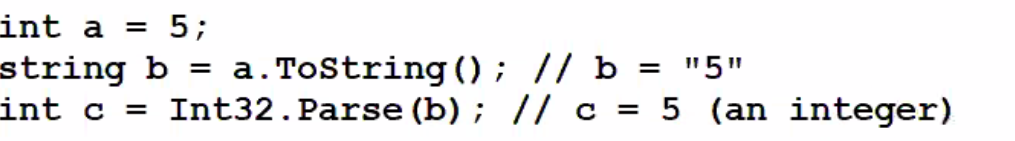
**Question 10**

How do you turn a string into an integer?

**Answer**

You can convert a string representation of a number to an integer using the following ways:

Parse() method, convert class, TryParse() method (recommended). The tryparse method converts the string representation of a number to its 16, 32, 64-bit signed integer equivalent. It returns Boolean which indicates whether the conversion succeeded or failed and so it never throws exceptions.



**Question 11**

Wat is the difference between precedence and associativity? Give an example where this makes a difference.

**Answer**

Precedence governs the order in which an expression’s operators are evaluated. Associativity is the direction (left or right) in which the operands of an operator are evaluated.

precedence

3 + 4 \* 5 = 23

3 + (4 \* 5) = 23

(3 + 4) \* 5 = 35

associativity

3 \* 4 / 5 = 12/5

(3 \* 4) / 5 = 12/5

3 \* (4 / 5) = 2.4

Precedence = vertical relationship

Associativity = horizontal relationship

**Question 12**

What is the definite assignment rule?

**Answer**

It is a rule simply stating that every variable must have a value before it’s read from.

**Question 13**

How are the prefix and postfix increment and decrement operators evaluate differently?

**Answer**

Increment operator increments by one. Decrement operator decrements by one. Post fix operator comes after the variable and evaluates the variable then modifies it. Prefix operator modifies first then evaluates it.

**Question 14**

What is string interpolation?

**Answer**

String interpolation renders many uses of the + operator obsolete for concatenating strings. It is more efficient than using the + operator; string concatenation using the + operator can be memory hungry by virtue of the way in which strings are handled by the .NET Framework. It is also arguably the readable and less error-prone. Where you can evaluate the variable within the context of a cmd.

String name = “Steven”;

Console.Write($”My name is {name}“); //prints: My name is Steven

Console.Write(“My name is {name}”); //prints: My name is {name} (The $ is a variable interpretation)

**Question 15**

What does the var keyword do?

**Answer**

It instructs the computer to figure out what the variable type is. It is useful to declare the implicitly-typed local variables without specifying an explicit type.