

## C# questions

- 1) What will the output be from this app.

The output will show: "location is null" and Returns the date as a string

```
0 references
class Program
{
    static String location;
    static DateTime date;

    0 references
    static void Main()
    {
        Console.WriteLine(location == null ? "location is null" : location);
        Console.WriteLine(date == null ? "date is null" : date.ToString());
    }
}
```

- 2) Write a function that takes a date of birth and returns their age.

```
static int DoB(string dateOfBirth)
{
    try
    {
        String[] currentDate = DateTime.Now.ToString("yyyy-MM-dd").Split("-");
        String[] dOB = DateTime.ParseExact(dateOfBirth, "yyyy-MM-dd", CultureInfo.InvariantCulture).ToString("yyyy-MM-dd").Split('-');

        var diffInYear = Int32.Parse(currentDate[0]) - Int32.Parse(dOB[0]);
        var diffInMonth = Int32.Parse(currentDate[1]) - Int32.Parse(dOB[1]);
        var diffInDay = Int32.Parse(currentDate[2]) - Int32.Parse(dOB[2]);

        return diffInMonth >= 0 && diffInDay >= 0 ? diffInYear : diffInYear - 1;
    }
    catch (Exception e)
    {
        Console.WriteLine("An error occurred. Please ensure you enter a valid date in the format yyyy-MM-dd \n\n" + e);
        return 0;
    }
}
```

- 3) Define method overloading with an example

Method overloading is when you have 2 functions with the same name that have different inputs. This can be the type or number of parameters specified. This can be used to perform different functions based on this. An example would be having an add function such as:

```
int add (int a, int b) {
    return a + B }

int add (int a, int b, int c) { return a + b + c
}
```

although both functions are add depending on the amount of arguments we put in so add(2,3) will call the first where as add(1,2,3) will call the second.

- 4) Define interface class in C# for reading a text file and then define a class that implements it.

A class that implements an interface class for reading text files would be StreamReader. An interface class defines the method and properties it must contain. For example: stream reader allows you to Read(), Close(), etc, as these are defined in the interface class beforehand.

- 5) The email also has zipped attachment with an MVC Model, View and Controller. The Model is Person.cs, the Controller is ExampleController.cs and the View is Index.cshtml. The code has been written very badly and so we'd like you to correct and improve the code and comment where you can to tell us what is happening. Consider the formatting in all pages and leave it in an acceptable and readable state.
- The View has a comment in with instructions for laying out the data. The data would represent the screenshots below.

## SQL questions

Below is a snapshot of a database table in SQL Server

	Column Name	Data Type	Allow Nulls
🔑	Id	int	<input type="checkbox"/>
	FirstName	nvarchar(100)	<input checked="" type="checkbox"/>
	LastName	nvarchar(100)	<input checked="" type="checkbox"/>
	Age	tinyint	<input checked="" type="checkbox"/>
	Country	nvarchar(50)	<input checked="" type="checkbox"/>
	Organisation	nvarchar(100)	<input type="checkbox"/>
▶		▼	<input type="checkbox"/>

And a snapshot of the data

	Id	FirstName	LastName	Age	Country	Organisation
	10000	John	Smith	56	Spain	JS Ltd
	10001	Jane	Johnson	29	France	
	10002	Paul	Bird	19	United Kingdom	
	10003	Mary	Lawson	37	France	
▶	10004	Jack	Goode	22	United Kingdom	Blah Co. Ltd
	10005	Anna	Maxwell	12	Portugal	
	10006	Nick	Carlton	66	Italy	
*	NULL	NULL	NULL	NULL	NULL	NULL

- 1) Write a SQL query that gets all first names starting with 'j' and order them by country.  
  
`SELECT * FROM Table WHERE FirstName LIKE 'j%' AND ORDER BY Country;`
- 2) Write a SQL query that gets all the results where the age is between 23 and 37 inclusive.  
  
`SELECT * FROM Table WHERE Age BETWEEN 23 and 37;`
- 3) Write a query to determine how many records there are where the country is United Kingdom  
  
`SELECT COUNT(*) FROM Table WHERE Country = "United Kingdom";`
- 4) Write a stored procedure to insert data into the table.

```
CREATE PROCEDURE insert_data
    @Id int,
    @FirstName nvarchar(100)=NULL,
    @LastName nvarchar(100)=NULL,
    @Age nvarchar(100)=NULL,
    @Country nvarchar(100)=NULL,
    @Organisation nvarchar(100)
AS
BEGIN
SET NOCOUNT ON INSERT INTO Table VALUES
    (
        @Id,
        @FirstName,
        @LastName,
        @Age,
        @Country,
        @Organisation )
END
GO
```