# PART 1

**Q1. What is delegate**

Delegate calls methods in order

1. Ref values updated between cells
2. Return value is the value of the last call

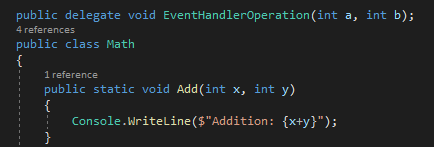
Delegate uses references to pass into methods with a set parameter and return type.

For my example I used EventHandlerOperation as my delegate and pass in two parameters which are 2 integers.

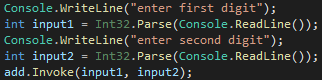


**Q2. What is Event**

Event handlers are methods that can be invoked through delegates and we can call our method by simply passing in parameters into the delegate which will inturn call the method that are invoked by the event.





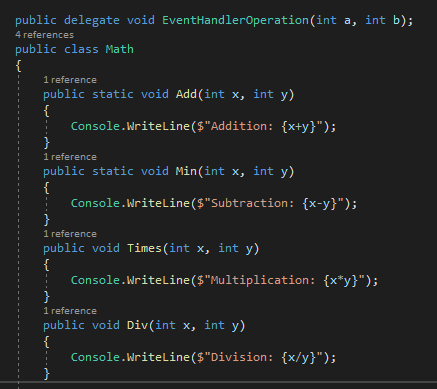


**Q3 Difference between delegate and event**

Delegate is mainly used to call methods which is similar to class while event is used to pass in parameters fields such as int, string etc. However, an event requires a delegate for it to pass the fields to while a delegate doesn’t require the help of event at all as it can simply be called and manually pass parameters into.

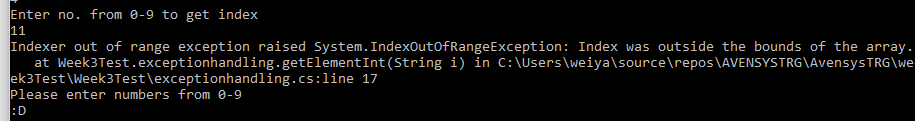
**Q4 How multicast delegate executes explain with example**

Multicast delegate is similar to method overloading but instead of method overloading, it simply runs all the functions that uses the same parameters. Multicast delegate simply executes by calling multiple functions with the same parameter.



**Q5 What is exception Handling**

Exception handling refers to catching exception errors which may cause the program to stop running as it has run into an exception. Instead of letting the program crash, we can use exception handling to display an error message to the user in the case of wrong input etc.



**Q6 Explain the order in try catch and finally**

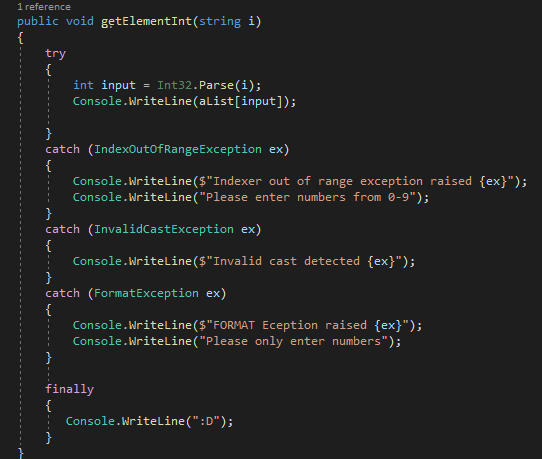
The program begins with TRY as it will try the input given by a user and move on to Catch any exception. If an exception is caught, it will display whatever is written in the code for it to display and after catching the exception, it will move on to finally and run the program there. If an exception wasn’t caught and the input is correct, it will move from try to finally block and run the code there.

IF EXCEPTION CAUGHT= TRY > CATCH > FINALLY

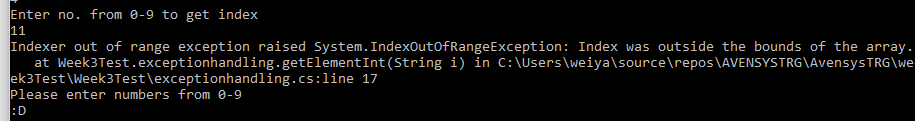
IF NO EXCEPTION Caught= TRY > FINALLY

**Q7 Explain with code example the FINALLY key word**

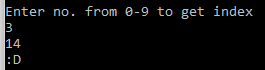
Finally, is used in a try catch and exception it is the last step of the method and will always display regardless of exception being caught. For my example, I have coded in a smiley face (:D) at the finally part of my program and every time it runs regardless of catching any exception the smiley face will be executed and displayed.



IF EXCEPTION CAUGHT= TRY > CATCH > FINALLY



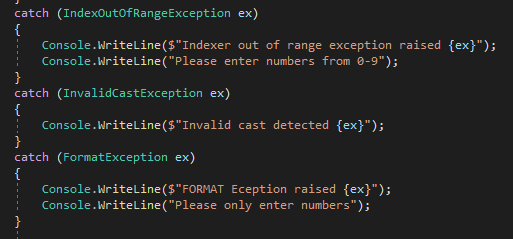
IF NO EXCEPTION Caught= TRY > FINALLY

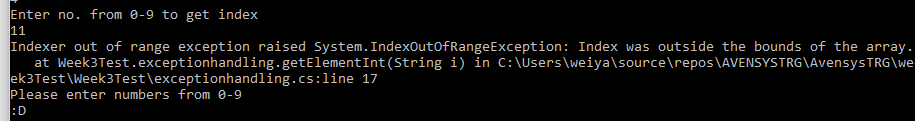


**Q8. Explain how a catch keyword works with example**

A catch keyword is used after the try part is completed. After the try code runs and the code try to convert a string to an INT if any errors were to occur, it will prompt an exception.

We can use the CATCH keyword with specific exceptions such as index out of bounds, format exception to prevent the program from crashing and display an error message to the user instead

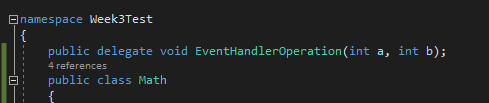




**Q9 How does a delegate handle an instance function and static function**

**Q10 How can a delegate be part of an interface declaration**

We can only use delegate to be part of an interface by using events with delegates or we can declare the delegate outside of the interface but within the same name space as delegates alone cannot be part of an interface.



**Q11 What are the type of different error scenarios and how are they handled by C# CLR to ensure program integrity**

1. Null reference exception
2. Invalid Operation Exception
3. OutOfRange Exception
4. Invalid cast exception
5. Format Exception

CLR will try and pass the values, run the code and if an error is found CLR will prompt the user that an exception has been identified.

# Part 2

1. A
2. A
3. C
4. D
5. D
6. C
7. B
8. D
9. C
10. B
11. D
12. A
13. A
14. C
15. B

# PART 4

**PERSON 1 to 1 CART 1 to many ITEMS**

ID Vege

Name MEAT

Login() Organise() Diary

Add2cart() LIST Items

Removeitem() Archieve()

PurchaseHist Pay()