1. **Расскажите? как генерируется исключение.**

Обычно система сама генерирует исключения при определенных ситуациях, например, при делении числа на ноль. Но язык C# также позволяет генерировать исключения вручную с помощью оператора throw. То есть с помощью этого оператора мы сами можем создать исключение и вызвать его в процессе выполнения.

1. **Расскажите методику обработки исключений.**

При использовании блока try...catch..finally вначале выполняются все инструкции в блоке try. Если в этом блоке не возникло исключений, то после его выполнения начинает выполняться блок finally. И затем конструкция try..catch..finally завершает свою работу.

Если же в блоке try вдруг возникает исключение, то обычный порядок выполнения останавливается, и среда CLR начинает искать блок catch, который может обработать данное исключение. Если нужный блок catch найден, то он выполняется, и после его завершения выполняется блок finally.

Если нужный блок catch не найден, то при возникновении исключения программа аварийно завершает свое выполнение.

1. **Какое ключевое слово служит для обозначения блока кода, в котором можно генерировать исключение?**

try

1. **Какие ключевые слова используются для обработки и генерации исключений? Расскажите об механизме обработке исключения?**

Try, Catch, finaly

1. **Что будет, если в программе нет предложения catch, способного обработать исключение?**

Аварийно завершится программа

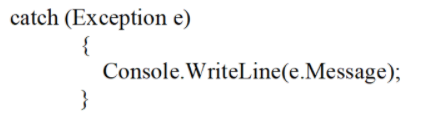
1. **Что такое фильтры исключения? Приведите пример**

Фильтры исключений позволяют обрабатывать исключения в зависимости от определенных условий. Для их применения после выражения catch идет выражение when, после которого в скобках указывается условие:

1. **Могут ли исключения быть вложенными?**

Да

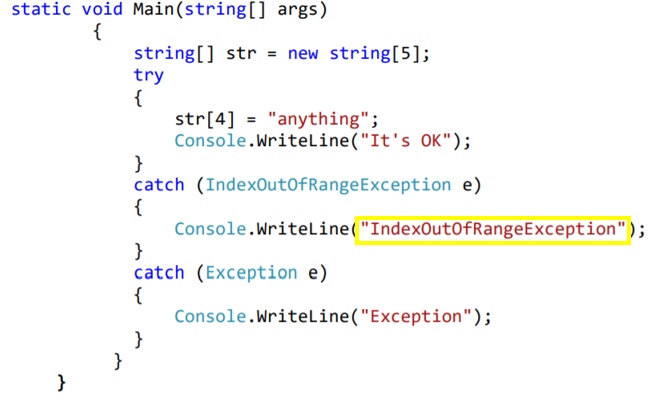
1. **Какой синтаксис нужно использовать в C# для отлова любого возможного исключения?**

****

1. **Чем следует руководствоваться при размещении обработчиков исключения?**

Обработчик исключения лучше всего размещать там, где может получиться ошибка.

1. **Что будет выведено на консоль в результате выполнения фрагмента листинга?**

****

1. **Как повторно сгенерировать то же самое исключение в блоке обработчике catch?**

Использовать Throw в catch

1. **Какие методы содержаться в классе Exception? Где и как их можно использовать?**

· *InnerException*: хранит информацию об исключении, которое послужило причиной текущего исключения

· *Message*: хранит сообщение об исключении, текст ошибки

· *Source*: хранит имя объекта или сборки, которое вызвало исключение

· *StackTrace*: возвращает строковое представление стека вызывов, которые привели к возникновению исключения

· *TargetSite*: возвращает метод, в котором и было вызвано исключение