1. ¿Qué es una excepción?

Las excepciones son tipos que se derivan en última instancia de System.Exception.

2. ¿Qué se coloca en el bloque “Catch”?

Si un bloque catch define una variable de excepción, puede utilizar dicho bloque para obtener más información sobre el tipo de excepción que se ha producido.

3. ¿Cómo construiría un objeto del tipo “Exception” personalizado?

Supongamos que tenemos una cuenta y un requerimiento funcional. El requerimiento funcional establece que si el saldo de la cuenta es inferior a cero se debe producir una excepción llamada “Saldo Negativo”. El entorno no provee esta excepción pues no en todos los sistemas esto sería considerado una situación no deseada.

Para poder construirla debemos generar una clase que herede de Exception. Es una buena práctica de programación que cuando una clase representa a una excepción, el nombre de la misma termine en Exception.

4. ¿Qué ocurre si en el bloque de código donde se produce la excepción el error no está siendo tratado?

Se ejecuta el finally.

5. ¿Cuál es el objeto de mayor jerarquía para el manejo de excepciones?

Si se coloca la excepción más genérica antes, nunca se llegaría a la más específica, pues la más genérica atraparía al error más específico. Esto se da pues las excepciones no escapan a la realidad de una relación jerárquica del tipo “es-un”.

6. ¿En qué namespace se encuentra la clase Exception?

System.Exception.

7. ¿Cuáles son las dos clases genéricas más importantes definidas en el FrameWork además de Exception?

8. ¿Qué instrucción se utiliza para poner en práctica el control e interceptar las excepciones?

Throw.

9. ¿Dónde se coloca el código protegido contra excepciones si se iniciara una excepción?

Dentro del finally.