**PREGUNTAS DE RESUMEN BIG BANG**

**ENTITY FRAMEWORK**

1. ¿Cuáles son los 3 tipos de entity framework?
2. Mencione tres ventajas de entity framework
3. ¿Cuáles son los 3 componentes de entity framework?
4. ¿Cuáles son las 4 reglas de entity framework?
5. ¿Cuáles son los 3 componentes de entity framework?

**JAVASCRIPT**

1. ¿JavaScript es compilado o interpretado?
2. Cuatro de las principales funciones de JavaScript son:
3. ¿Qué pasa si se añade un número y una cadena?
4. ¿Qué hace ´confirm´ y ´prompt´?
5. Mencione y explique 5 eventos de JScript

**JQUERY**

1. Defina Jquery
2. ¿En qué se diferencia de JavaScript?

1. ¿Qué función reemplaza el símbolo $()?
2. ¿Cómo agrega color al fonde de una página?

1. ¿Cómo oculta un elemento?
2. ¿Tres ventajas de Jquery son?

1. ¿Qué es el DOM?

1. Seis métodos de ejemplo de Jquery con los que se puede manipular CSS son:

**MVC**

1. ¿Qué es MVC?
2. Explique sus partes
3. Cuatro ventajas de MVC son:

1. Dos desventajas de MVC son:
2. ¿Con MVC se tiene un control total sobre como las URLs serán formateadas o mostradas en el navegador? Explique su respuesta.

**ORDENAMIENTOS**

1. ¿Qué es un ordenamiento?
2. ¿Cuál es su propósito principal?
3. ¿Cuándo un ordenamiento es eficiente?
4. ¿Cuándo un ordenamiento es interno? Dé 2 ejemplos

**PATRONES DE DISEÑO**

1. ¿Qué son patrones de diseño?
2. ¿Son exclusivos de la programación orientada a objetos?
3. ¿Cuáles son los 3 tipos de Patrones de diseño que hay?

1. Mencione y explique 7 de las más importantes

1. ¿Cuáles son los 4 elementos de un Patrón de diseño?

**PROGRAMACION ORIENTADA A OBJETOS**

1. ¿Qué es la POO?
2. ¿Cuáles son los 7 componentes principales de POO? Explíquelos (Herencia, acoplamiento y cohesión, ocultamiento y encapsulamiento, abstracción, y polimorfismo)

1. ¿Qué es Clase y Objeto?
2. ¿Qué son atributos Estáticos (o Shared en VB)?
3. ¿Qué es Garbage Collector? ¿Cuándo se ejecuta? ¿Se puede ejecutar manualmente? ¿Cómo?
4. Respecto a ocultamiento: mencione y explique 5 modificadores de acceso.
5. Respecto a polimorfismo, mencione y explique 3 tipos.

1. ¿Qué es herencia simple?
2. Respecto a herencia, ¿qué es herencia con sobre escritura? ¿Cómo se implementa en VB.NET?
3. ¿Qué es shadowing?
4. ¿En qué se diferencia shadowing de herencia? Dé un ejemplo\*\* <https://www.c-sharpcorner.com/article/difference-between-shadowing-and-overriding-in-oop/>

44. Respecto a herencia, si un método no se declara como overridable (en clase abstracta o no, es indiferente), ¿se puede heredar? ¿qué diferencia hace declararlo como overridable?\*\* <https://www.thoughtco.com/overrides-in-vbnet-3424372>

1. ¿Cómo hago una referencia explícita a un método o propiedad de la clase base?
2. Respecto a una clase abstracta: defínala; ¿se puede instanciar? ¿Qué es lo único que puede hacerse con ella?
3. ¿Cómo se declara en C# y como en VB.NET; pueden tener constructor? ¿Desde dónde se puede invocar? ¿Puede tener propiedades y métodos? deben ser obligatoriamente sobrescritos? Explique.
4. ¿En qué se diferencia la clase abstracta de las interfaces?
5. Si deseamos que una clase no pueda ser heredada, ¿Cómo debe declararse?
6. ¿Puede una clase abstracta tener métodos implementados (no solo nombre, parámetros y tipo de retorno, sino funcionalidad)?\*\*   
   <https://www.dotnetperls.com/mustinherit-vbnet>
7. ¿Qué definen los Campos o Miembros de clase?
8. ¿Qué es una Propiedad?

1. ¿Qué es el Constructor? Como se declara en VB?
2. ¿Puede una clase tener varios constructores? Explique.
3. ¿Qué pasa con el compilador si declaramos un constructor con parámetros?
4. ¿Cómo se define un método que devuelve parámetros y el que no lo hace?
5. ¿Qué es un método abstracto? ¿Cómo se declaran en C# y en Visual Basic? ¿Se pueden declarar fuera de clases abstractas?
6. ¿Los atributos privados de una clase son accesibles por sus clases derivadas? Si no es así, ¿cómo son accesibles?
7. ¿Qué es sobrecarga y que permite respecto a constructores?
8. ¿Qué se necesita para que una sobrecarga sea válida respecto a número o tipo de parámetros?
9. ¿Se pueden declarar dos métodos con el mismo nombre y además con el mismo tipo y cantidad de parámetros?
10. ¿Cuál es la diferencia entre ByVal y ByRef?
11. ¿Qué modificadores de acceso para controlar la visibilidad tienen las clases y sus elementos por defecto?
12. Describa las 5 partes de SOLID con ejemplos para cada una.

1. Describa cuales serían 7 buenas prácticas de código mantenible.

**SCRUM**

1. ¿Qué es SCRUM? ¿Es un proceso o técnica? Sino... ¿Qué es?
2. ¿En qué consiste el marco de trabajo de Scrum? Mencione las 4 partes.

1. ¿Es SCRUM una metodología de gestión de proyectos o reemplazar a las mismas?
2. ¿Cuáles son las 4 tareas de cada iteración?
3. Mencione y detalle los 3 roles de Scrum.
4. ¿Qué es un Sprint? ¿Cuánto dura? ¿Se inicia el sgte de inmediato? ¿Se puede considerar como un proyecto?
5. Describa las 4 distintas reuniones de Sprint y sus características.

1. Describa una Tarjeta de Product Backlog.

**INVEST**

1. ¿Qué es INVEST? Describa cada letra.
2. Si una historia de usuario no cumple alguno de los puntos de INVEST, ¿qué debería hacerse?
3. ¿Qué es Disponibilidad del equipo?

77. ¿Qué es Velocidad del equipo?

1. Cuatro diferencias entre desarrollo incremental y desarrollo iterativo son:

**TDD y Refactoring**

1. Mencione los 2 niveles de TDD.
2. De 3 características de ATDD y 2 para TDD.
3. Cuatro aspectos a considerar sobre TDD son:

1. Dé cinco características sobre refactorización.

**CASOS DE USO**

1. Mencione 3 características de Casos de Uso.

**SQL**

-Entidad relación

84. ¿Para qué se usa el modelo ER?

1. Explique sus 3 partes fundamentales.

1. ¿Qué es la cardinalidad?

87. Dé ejemplos de 3 formas de representar la cardinalidad.

1. Cuatro maneras de optimización de consultas SQL son:
2. ¿Qué es un Plan de Ejecución. Dé dos características.
3. ¿Qué debe revisarse al Analizar el plan de ejecución?
4. ¿Que son joins? Explique con ejemplos los 3 tipos.

-Transacciones en SQL Server

91. ¿Qué es una transacción?

1. ¿A qué se refiere el termino Commit?
2. ¿A qué se refiere el termino Rollback?
3. Detalle cuatro términos de transacciones

-ACID

95. Defina qué significa cada letra en ACID

1. ¿Qué controlan los niveles de aislamiento?
2. ¿Qué se puede hacer para asegurar la integridad de las operaciones?
3. ¿Qué puede causar un bloqueo excesivo de datos?
4. ¿Qué es un deadlock? Dé dos maneras de evitarlos.

1. ¿Qué son blocklocks? ¿Qué relación tienen con los Bloqueos de datos?
2. Explique los dos tipos de Bloqueo de datos.
3. Explique los cinco niveles de aislamiento.
4. ¿Qué se puede hacer con un GroupBy?
5. ¿Cuáles son cinco aspectos a considerar respecto a GroupBy?
6. ¿Qué es un índice?
7. Explique los 2 tipos de Índice

**VISUAL BASIC**

-Expresiones lambda

107. Defina expresiones lambda

1. ¿En lugar de qué pueden usarse? ¿Por qué?\*\*
2. ¿Qué ahorra usar expresiones lambda? ¿Qué se evita con ellas?
3. ¿Cuáles son las dos maneras de declararlas y por qué? ¿Qué se puede omitir y bajo qué condiciones?
4. ¿Qué parámetros no acepta?
5. ¿Cuándo un delegado hace match con cualquier método?
6. ¿Qué debe coincidir entre una función lambda y un delegado? ¿Qué no es necesario que coincida y porque?

-Declaración inline de delegados

113. ¿Cómo se declaran cuando retorna un valor? ¿Cómo cuando no lo hace? Escriba un ejemplo para cada caso. ¿Qué sucede con los nombres y tipos de argumentos?

1. ¿Acepta parámetros genéricos?

-Delegados en Colecciones de datos

115. Explique cuatro ejemplos de uso de delegados en colecciones de datos.

1. Dé la relación entre funciones lambda y entity framework.

-Lambda en MVC

117. ¿Cuál es la relación entre lambda y MVC? Dé dos ejemplos.

-Programación Concurrente en VB.NET

118. Defina TPL. ¿Qué hace? ¿En cuáles APIs están sus ensamblados?

119. Describa cuatro características de TPL.

120. ¿Qué es Paralelismo de Datos?

121. ¿Qué clase y métodos usa?

122. ¿Qué hace TPL al invocar estos métodos?

123. ¿Qué hace el calendarizador de tareas?

124. Describa los métodos For, ForEach e Invoke para la clase Parallel, con sus semejanzas y diferencias.

125. Defina Task. ¿Cuáles son las dos formas de declararlo? Explique las tres maneras de iniciar tareas asíncronas con la clase Task.

126. ¿Cuáles son las dos clases que son la base de la Programación Asíncrona?

127. ¿Qué pasa con la UI y el flujo de la aplicación cuando se utilizan métodos asíncronos?

128. ¿Con qué palabra se definen los métodos asíncronos?

129. ¿Qué sucede al usar Await dentro de un método Asíncrono? ¿Qué pasa si no existe ningún Await en un método Asíncrono?

* Dentro de un método Asíncrono, ¿se puede llamar un método NO Asíncrono?
* Dentro de un método Asíncrono, ¿se puede llamar otro método Asíncrono?
* Dentro de un método Asíncrono, ¿cuál es la diferencia entre llamar un metido asíncrono y otro que no lo es? \*\*  
    
    
    
   https://blog.stephencleary.com/2012/02/async-and-await.html

130. ¿Qué es Task y Task(Of T)? ¿En que se diferencian?

131. ¿Se puede usar Await dentro de un método que no está definido como Asíncrono?\*\*

Ver seccion "Async and Await are a Pair" en: https://richnewman.wordpress.com/2012/12/03/tutorial-asynchronous-programming-async-and-await-for-beginners/

132. Dentro de un método Asíncrono, al llamar otro método (ya sea Asíncrono o no), cual es la diferencia de llamarlo usando Task vs Await? De un ejemplo para cada caso.\*\* (Para responder esta pregunta se puede correr el código en la presentación de BigBang "Introducción a la Programación Concurrente en VB", y llamar al método LongMethodAsync() de las dos maneras:

Dim myTask As Task(Of DateTime) = LongMethodAsync()

Dim myTask = Await LongMethodAsync()

)

-Manejo de excepciones

133. ¿Cómo se manejan las excepciones en métodos invocados con Await?

134. ¿Hay alguna diferencia en la manera de manejar excepciones entre métodos Síncronos y Asíncronos?\*\* http://www.informit.com/articles/article.aspx?p=2429291&seqNum=6

135. ¿Puede ir Await dentro de un bloque Catch o Finally en VB.NET? (Parece que si es posible en C#: <https://blog.stephencleary.com/2014/06/await-in-catch-and-finally.html>)

136. Mencione y explique tres malas prácticas y errores comunes.

**INYECCION DE DEPENDENCIAS**

137. ¿Qué es IoC? Explique. ¿Quién crea los objetos consumidos?

138. ¿Qée es Inyección de Dependencias?

139. ¿Para qué se utiliza? ¿Como se logra en .NET?

140. ¿Cuál es la relación entre IoC y DI?

141. ¿Qué debe tener una clase para ser inyectable? ¿Cuál puede ser un problema? ¿Como se resuelve?

142. ¿Qué es un Contenedor de IoC? ¿Qué administra? ¿En qué principios se basa?

143. ¿Qué inicializa un IoC? ¿Qué gana la clase consumidora utilizando un Contenedor de IoC?

**UNITY**

144. ¿Qué es Unity?

145. ¿Cuáles son las dos maneras de especificar las dependencias en Unity?

146. ¿Cómo se agrega Unity a MVC? ¿En qué archivo se configura? ¿En que ruta queda dicho archivo?

147. ¿Qué acción adicional debe hacerse al proyecto de pruebas al hacer pruebas unitarias usando Unity?

148. ¿Qué se puede lograr con configuraciones avanzadas de Unity?

149. ¿Qué permite Unity para crear instancias que requieran de argumentos adicionales en su constructor?

150. ¿Qué puede hacer el repositorio Mock? Detalle dos opciones.

151. ¿Cuál es la ventaja de tener un archivo de configuración? Mencione dos ventajas.

152. ¿Para qué sirve el método Resolve de la clase UniyContainer?

153. ¿Qué se hace si no se quiere inyectar dependencias en el constructor?

**PILA**

154. ¿Cuándo se usa una Pila? ¿Qué significa LIFO? Dé un ejemplo práctico en el que se puede usar una pila.\*\*

**COLA**

155. ¿Qué es una cola? ¿Cómo funciona? ¿Qué significa FIFO? De un ejemplo práctico de una cola \*\*

**MICROSERVICIOS**

156. ¿Qué son Microservicios?

157. ¿Qué relación tienen los distintos servicios en este estilo arquitectural?

158. Mencione cinco características de Microservicios.

159. ¿Qué es común al trabajar con Microservicios? ¿Es obligatorio?

**VERBOS DE HTTP**

160. Describa los cinco verbos existentes.

161. Liste y describa ocho códigos de respuesta estandarizados.

**SERIALIZACION**

162. Defina Serialización.

**DEBUG REMOTO**

163. Describa los cinco pasos necesarios para hacer un debug remoto.

**BLOQUE USING**

164. ¿Para qué se utiliza el bloque Using?

165. ¿Cuándo se usa Using en que momento se liberan los recursos?

166. ¿Qué interfaz es necesaria en quien consume los recursos? ¿Qué garantiza?

167. Ante un error, ¿cómo se comporta el bloque Using? ¿Cómo se obtendría el mismo comportamiento usando Try y Catch?

168. ¿Cómo afecta el scope de Using a las variables declaradas en él?

169. ¿Using debe tener poco código?

**GAC GLOBAL ASSEMBLY CACHE**

170. ¿Cuáles son 4 características del GAC?

171. Dos puntos respecto a Registrar DLLs en el GAC.

172. Tres puntos respecto a eliminar DLLs del GAC.

173. Dos puntos para ver contenido del GAC.

174. ¿Qué debe hacerse para eliminar el caché de ASP.NET?

**PRUEBAS DE ESTRES**

175. Dos Características de funciones testeables son:

176. ¿Cuáles son los dos tipos de prueba que existen? Descríbalas y explique porqué son diferentes entre ellas.

177. Dos maneras en que Load Test hace ambas pruebas:

178. ¿Qué es sample rate? ¿Qué valores se usan para pruebas cortas y que valores para pruebas largas? ¿Por qué?

179. Dos maneras de configurar una prueba son:

180. ¿Qué es el tiempo de calentamiento y para qué se usa?

181. ¿Para qué se usa 'el perfil de tiempo para pensar'?

182. ¿Qué es una carga constante y qué una carga en pasos?

183. ¿Cuántas pruebas unitarias o web test pueden agregarse a una prueba de carga?

184. ¿Cuál es la diferencia entre UNIT TESTs y Web Tests?\*\*

185. ¿Qué es un Text Mix Model? ¿Cuáles son las cuatro maneras de realizarlo? Explíquelas

186. ¿Qué es Network Mix Model? Dé dos características

187. ¿Qué permiten los 'counter sets'? ¿Cuáles son los tres contadores a recolectar?

**EXCEPCIONES**

188. ¿Qué es una excepción?

189. ¿Qué hay en .NET para cada aplicación acerca de las excepciones?

190. ¿Cuál es la base de todas las expeciones? ¿De quién hereda esta base?

191. Mencione y explique las tres propiedades de la base de todas las excepciones.

192. ¿Qué es 'Data'?

193. Explique las tres partes de una excepción.

194. Mencione y explique las tres maneras de hacer un reenvío de excepciones.

195. Menciones seis Errores comunes al usar el manejo de excepciones.

**POSTBACK**

196. ¿Qué es Postback? Explíquelo con cuatro caracteristicas.

**RAZOR**

197. ¿Qué es Razor? Explíquelo con cuatro caracteristicas.

**HERENCIA**

198. ¿Cómo se declara una clase que hereda de otra?

199. ¿Qué debe hacerse si se quiere evitar que una clase sea heredable?

200. ¿Una clase que hereda a otra está obligada a sobrescribir todos sus métodos?

201. ¿Una clase que implementa varias interfaces está obligada a implementar todos los métodos de la interfaz?

202. ¿Una clase abstracta se puede instanciar?

203. ¿Para qué se utiliza ForEach, Next? ¿Puede usarse en colecciones que NO implementen la interfaz IEnumerable?\*\*

204. ¿Qué operador se usa para concatenar strings?

205. Si se concatena un string y un número, ¿qué sucede si usamos '+' y que sucede si usamos '&'? Dé un ejemplo para cada caso.\*\*

206. ¿Qué pasa con el objeto original de un string cuando se modifica?

207. ¿Cuándo se usa String y cuando StringBuilder?

208. Explique ocho métodos utilitarios del objeto string.

**CLASE GENERICA**

209. Defina Clase Genérica. ¿Cómo se declara? ¿Por qué es fuertemente tipada?

210. ¿Por qué usar Clase Genérica en vez de Object? \* <https://docs.microsoft.com/en-us/dotnet/visual-basic/programming-guide/language-features/data-types/generic-types>

**WCF**

211. ¿Qué significan las letras de WCF?

212. ¿Qué es WCF?

213. Mencione y explique tres patrones de mensajería que se soportan en WCF.\*\*

214. Menciones dos aspectos de seguridad existentes en WCF.

215. ¿Qué tipo de transacciones soporta? Explique cada una \*\*

216. ¿WCF soporta Ajax y REST?

217. Escriba un ejemplo de archivo de configuración de WCF completo, con todas sus partes.

218. ¿Qué muestra los mensajes de error detallados al cliente para conocer y depurar el error? ¿En qué tipo de ambientes se usa y porque?

219. Si un archivo de configuración escucha Critical y se lanza un Error, ¿qué sucede? ¿Cómo se resuelve?

220. ¿Por medio de qué se da la comunicación entre clientes y servicios?

221. Defina EndPoint. Describa sus tres partes.

222. Describa dos componentes adicionales de WCF.

223. Explique en detalle cuatro tipos de Binding.

224. Dos posibles protocolos de transporte de WCF son:

225. ¿Tres tipos de Encoding son? Explique cada uno \*\*

226. ¿Qué permite configurar los Binding? 4 partes

227. Explique los tipos de Manejo de instancia en WCF

228. Explique los tipos de Manejo de concurrencia

229. Explique los tipos de sesión en el IIS

230. ¿Es lo mismo los tipos de sesión en el IIS que tipos de sesión en ASP.NET? ¿Por qué? Explique su respuesta \*\*

231. ¿En que archivo se configura el Behavior? ¿Para limitar qué se puede configurar un Behavior?

232. ¿Qué sección no debe olvidarse al declarar un behavior? ¿Por qué? \*\*

233. ¿Dentro de qué sección va el Bindings y Behaviors en el web.config?

234. ¿Qué sección debe agregarse dentro de system.serviceModel para configurar el endpoint en web.config?(CREO QUE LA RESP ESTA MAL)

235. ¿En que sección va el Behavior y en cual el Binding?

236. ¿Qué se usa para probar la accesibilidad de un endpoint? Describa los pasos para su uso\*\*

237. ¿Cómo debe realizarse la transferencia de mensajes en un WCF?

238. ¿Cuáles son los tres puntos a considerar para tener un WCF seguro?

239. ¿En qué parte del ABC se configura el modo de seguridad?

240. Explique los cinco tipos de seguridad de un WCF.

241. Explique en detalle cual es la diferencia entre Seguridad Mixta y Both? \*\*

http://www.wcftutorial.net/Transfer\_Security\_Mode.aspx

https://medium.com/@mirzafarrukh13/wcf-security-97945377cfa0

https://www.tutorialspoint.com/wcf/wcf\_security.htm

<http://www.silverlighthack.com/category/WCF.aspx>

242. ¿Cuáles son los dos puntos en los que se basan los modos de seguridad de WCF?

243. Describa los cuatro modos de seguridad en WCF.

244. ¿En qué consiste el Tracing?

245. ¿Qué hace el listener? Cuatro distintas maneras de guardar la información son:

246. ¿Para que se usa la información obtenida?

247. ¿En qué archivo y en qué sección se configura el tracing?

248. ¿Qué diferencia al tracing del debug?

249. ¿Cuáles son los cinco distintos niveles de trazabilidad existentes? ¿Puede encontrarse la misma información en niveles distintos? ¿Por qué?

250. ¿Cuántos listener puede tener un servicio? ¿Se pueden personalizar?

251. Describa seis opciones de logueo de mensajes

252. ¿Se pueden guardar el tracing en una ruta específica? ¿Porque es necesario un formateador de XML?

253. ¿Para qué sirve el Autoflush?

254. ¿Qué problema puede ocurrir si se usa un archivo para el log si existe mucha concurrencia?

255. ¿Que es el web.config? ¿Dónde se ubica?

256. ¿Qué es el machine.config? ¿Puede haber más de uno en una sola máquina? ¿A cuáles aplicaciones se aplica su configuración? ¿Cuál es su diferencia respecto al Web.config?

257. Tres caracteristicas de WCF Restful son:

258. Explique en detalle la diferencia entre WCF y WebServices.

259. ¿Cuál es la diferencia entre SOA y SOAP?

260. ¿Cuál es la diferencia entre SOAP y REST?

**HTML5**

261. ¿Qué implementa que antes dependía de algo más?

262. Explique cuatro caracteristicas que un sitio debe tener:

263. ¿En que sección se puede declarar el <script>? ¿Dónde se recomienda hacerlo? ¿Cuáles son dos maneras de agregar código?

264. Mencione las 13 partes en las que puede dividirse un HTML? Haga un gráfico con la distribución de las 6 principales.

265. Defina CSS. ¿Qué hace?

266. ¿Cuál es la característica más importante de CSS y porque?

267. ¿Qué hace el modelo de cajas? ¿Cuáles son sus cuatro componentes?

**XML**

268. ¿Cuál es la relación entre XML y SGML? ¿Qué es SGML? \*\*

269. ¿Qué es XML?

270. Mencione y detalle diez características de XML

271. ¿Qué hace un DTD? ¿Qué se comprueba con él?

272. ¿Qué es XSD? ¿A qué sustituye y por qué?

273. ¿Qué es XSLT?

274. Dé cuatro ejemplos de conversiones posibles con XSLT.

274. ¿Qué es DTSX?