

# Examen 98-361

## TA: Fundamentos de Desarrollo de Software

---

### Diseño de exámenes.

MTA es una nueva certificación bajo el programa de certificación de Microsoft que valida el conocimiento fundamental necesario para comenzar la construcción de una carrera en las tecnologías de Microsoft.

Los candidatos para este examen obtendrán una certificación de MTA así como el acceso a los beneficios del programa de certificación de Microsoft.

Se recomienda que los candidatos estén familiarizados con los conceptos y tengan experiencia práctica con las tecnologías descritas aquí, ya sea tomando cursos de formación o trabajando con tutoriales y las muestras disponibles en MSDN<sup>®</sup> y Microsoft<sup>®</sup> Visual Studio<sup>®</sup>.

### Dominio de Objetivos

#### 1. Comprendiendo la programación básica

1.1. Comprender el almacenamiento computacional y los tipos de datos.

*Este objetivo puede incluir más no se limita a:* cómo la computadora almacena programas e instrucciones en la memoria; pila basada en la asignación de memoria y arboles; requerimientos de tamaño de la memoria para varios tipos de almacenamiento de datos; datos numéricos y datos de texto.

1.2. Comprender las estructuras de decisión computacionales.

*Este objetivo puede incluir más no se limita a:* varias estructuras de decisión utilizadas en todos los lenguajes de programación computacional; estructuras de decisión IF; estructuras de decisión múltiple tales como If...Else y switch/Select Case; lectura de diagramas de flujo; tablas de decisión; expresiones de evaluación

1.3. Identificar el método apropiado el manejo de la repetición.

*Este objetivo puede incluir más no se limita a:* bucles For, bucles While, bucles Do..While, y recursividad.

1.4. Comprender el manejo de errores.

*Este objetivo puede incluir más no se limita a:* manejo de excepciones estructuradas

## **2. Comprender la Programación Orientada a Objetos**

2.1. Comprender los conceptos básicos de las clases.

*Este objetivo puede incluir más no se limita a:* propiedades; métodos; eventos; y constructores; como crear una clase; como utilizar clases en código

2.2. Comprender la herencia.

*Este objetivo puede incluir más no se limita a:* heredar la funcionalidad de una clase base a una clase derivada

2.3. Comprender el polimorfismo.

*Este objetivo puede incluir más no se limita a:* extender la funcionalidad en una clase después de heredar de una clase base; sobre escritura de métodos en la clase derivada

2.4. Comprender el encapsulamiento.

*Este objetivo puede incluir más no se limita a:* creación de clases que ocultan sus detalles de implementación mientras aun permiten el acceso a la funcionalidad requerida a través de la interfaz; modificadores de acceso

### **3. Comprender el Desarrollo General de Software**

3.1. Comprender la administración del ciclo de vida de la aplicación.

*Este objetivo puede incluir más no se limita a:* fases del ciclo de vida; evaluación de software

3.2. Interpretar las especificaciones de la aplicación.

*Este objetivo puede incluir más no se limita a:* lectura e interpretación de las especificaciones de la aplicación en: prototipos, código y componentes.

3.3. Comprender los algoritmos y las estructuras de datos.

*Este objetivo puede incluir más no se limita a:* arreglos, pilas, colas, listas vinculadas, y algoritmos de ordenamiento; implicaciones de desempeño de varias estructuras de datos; selección de la estructura de datos adecuada.

No: Análisis de algoritmos

### **4. Comprender las aplicaciones Web**

4.1. Comprender el desarrollo de una página Web.

*Este objetivo puede incluir más no se limita a:* HTML, Hojas de Estilo en Cascada (CSS), JavaScript

4.2. Comprender el desarrollo de aplicaciones en Microsoft ASP.NET Web.

*Este objetivo puede incluir más no se limita a:* ciclo de vida de una página; modelo de eventos; administración de estados; aplicaciones en cliente vs. programación de aplicaciones en servidor

4.3. Comprender el alojamiento Web.

*Este objetivo puede incluir más no se limita a:* creación de directorios virtuales y sitios Web, uso de aplicaciones Web; comprender el papel de los Servicios de Información de Internet

4.4. Comprender los Servicios Web.

*Este objetivo puede incluir más no se limita a:* servicios Web que se consumirán por aplicaciones cliente; acceso a servicios Web desde una aplicación cliente; SOAP y Web Service Definition Language (WSDL)

## **5. Comprender las Aplicaciones de Escritorio**

### 5.1. Comprender las aplicaciones Windows® Forms.

*Este objetivo puede incluir más no se limita a:* modelos de eventos en Windows Forms; herencia visual; diseño de interfaz de usuario UI; uso de aplicaciones Interfaz de Documentos Múltiple (MDI) e Interfaz de Documento Únicos (SDI)

### 5.2. Comprender las aplicaciones de consola.

*Este objetivo puede incluir más no se limita a:* características y capacidades de aplicaciones de consola

### 5.3. Comprender Windows Services.

*Este objetivo puede incluir más no se limita a:* características y capacidades de Windows Service

## **6. Comprender las Bases de Datos**

### 6.1. Entender los sistemas de administración de bases de datos relacionales.

*Este objetivo puede incluir más no se limita a:* características y capacidades de productos de bases de datos; diseño de base de datos; Diagramas Entidad-Relación (ERDs); conceptos de normalización

### 6.2. Entender los métodos de consultas de bases de datos.

*Este objetivo puede incluir más no se limita a:* lenguaje de consulta estructurado (SQL), creación y acceso a procedimientos almacenados, actualización de datos, selección de datos

### 6.3. Comprender los métodos de conexión de bases de datos.

*Este objetivo puede incluir más no se limita a:* conexión a varios tipos de almacenamiento de datos tales como archivo plano; archivos XML; objeto en-memoria; optimización de recursos