

OBJETIVOS DEL EXAMEN DE IT SPECIALIST



Python

1. Operaciones usando tipos de datos y operadores

1.1 Evaluar expresiones para identificar los tipos de datos que asigna Python Variables

- str, int, float y bool

1.2 Realizar operaciones de datos y tipos de datos

- Conversión de tipos de datos, indexación, división, construcción de estructuras de datos

1.3 Determinar la secuencia de ejecución en función de la precedencia del operador

- Asignación, comparación, lógica, aritmética, identidad (es), contención (en)

1.4 Seleccionar operadores para lograr los resultados previstos

- Asignación, comparación, lógica, aritmética, identidad (es), contención (en)

2. Control de Flujo con Decisiones y Bucles

2.1 Construya y analice segmentos de código que usen sentencias de bifurcación

- if, elif, else, expresiones condicionales anidadas y compuestas

2.2 Construya y analice segmentos de código que realicen iteraciones

- while, for, break, continue, pass, loops anidados, loops que incluyen compuestos expresiones condicionales

3. Operaciones de entrada y salida

3.1 Construir y analizar segmentos de código que realizan la entrada de archivos y operaciones de salida

- Abrir, cerrar, leer, escribir, agregar, verificar existencia, eliminar, con declaración

3.2 Construir y analizar segmentos de código que realizan entrada de consola y operaciones de salida

- Lea la entrada desde la consola, imprima el texto formateado (método string.format(), método f-String), use argumentos de línea de comando

4. Documentación y estructura del código

4.1 Segmentos de código de documento

- Use sangría, espacios en blanco, comentarios y cadenas de documentación; generar documentación usando pydoc

4.2 Construya y analice segmentos de código que incluyan funciones. definiciones

- Firmas de llamada, valores predeterminados, retorno, definición, paso

5. Solución de problemas y manejo de errores

5.1 Analice, detecte y corrija segmentos de código que tengan errores

- Errores de sintaxis, errores de lógica, errores de tiempo de ejecución

5.2 Analice y construya segmentos de código que manejen excepciones

- Probar, excepto, si no, finalmente, aumentar

5.3 Realizar pruebas unitarias

- Unittest, funciones y métodos

6. Operaciones usando Módulos y Herramientas

6.1 Realice operaciones básicas mediante el uso de módulos incorporados

- Matemáticas, fecha y hora, io, sys, os, os.path, aleatorio

6.2 Resuelva problemas informáticos complejos mediante el uso de módulos integrados

- Matemáticas, fecha y hora, aleatorio