

Programación I
Grado en
Análisis de Negocio



UNIVERSIDAD
NEBRIJA

GUÍA DOCENTE

Asignatura: Programación I

Titulación: Grado en Análisis de Negocio

Carácter: Obligatorio

Idioma: Castellano

Modalidad: Presencial

Créditos: 6

Curso: 1º

Semestre: 2º

Profesores/Equipo Docente: García Maezo, Rubén

1. COMPETENCIAS Y RESULTADOS DE APRENDIZAJE

1.1. Competencias

Competencias básicas: CB1, CB2, CB3, CB4, CB5

Competencias generales: CG1, CG2, CG5, CG6, CG11, CG12, CG13, CG14, CG15, CG16, CG17, CG19

Competencias específicas: CE8, CE9, CE18, CE21, CE22, CE23, CE24

1.2. Resultados de aprendizaje

- Comprender los fundamentos de la programación y su aplicación en el análisis de negocios.
- Aprender sobre tipos de datos, operadores básicos, y estructuras de control en la programación.
- Desarrollar habilidades en la manipulación de tablas, cadenas, y estructuras de datos.
- Entender y aplicar instrucciones de control, funciones, condicionales y ciclos en la resolución de problemas.
- Mejorar la eficiencia en la programación y el manejo de archivos de texto.
- Comprender la estructura de un programa y su implementación en contextos de negocio.

2. CONTENIDOS

2.1. Requisitos previos

No hay requisitos previos

2.2. Descripción de los contenidos

El curso de "Introducción a la Programación" en el contexto de Business Analytics aborda de manera integral los fundamentos de la programación, necesario para entender cómo desarrollar y aplicar soluciones de software eficaces en el análisis de negocios. El programa enfatiza la importancia de los principios básicos de programación y su aplicación directa en entornos empresariales, asegurando que los estudiantes adquieran una base sólida para manejar diversos desafíos técnicos.

Los módulos del curso cubren un amplio espectro de temas relacionados con la programación. Desde el estudio de tipos de datos y operadores básicos, que son fundamentales para manejar

datos y ejecutar operaciones lógicas y matemáticas, hasta la manipulación y gestión de strings y estructuras de datos complejas, indispensables en el manejo de grandes volúmenes de información en negocios.

Asimismo, se enseñarán técnicas para utilizar instrucciones de control que dirigen el flujo de ejecución de programas, habilidades requeridas para la resolución lógica de problemas de programación. El curso también incluye una inmersión profunda en cómo definir y utilizar funciones y procedimientos para modularizar el código, mejorar la reusabilidad y simplificar la estructuración de programas complejos.

En cuanto a las estructuras condicionales y los bucles, se explicará su uso para realizar tareas repetitivas y tomar decisiones programáticas, aumentando así la eficiencia en la resolución de problemas específicos del ámbito empresarial. Además, se abordarán técnicas avanzadas para optimizar la eficiencia del código y el manejo de archivos de texto, competencias necesarias para el procesamiento y análisis de datos en formatos variados.

Finalmente, el curso destacará la aplicabilidad de estas habilidades de programación en una variedad de industrias, como finanzas, retail y tecnología. Se discutirá la relevancia de la programación en roles de análisis de datos e inteligencia empresarial, resaltando cómo la programación puede mejorar significativamente la capacidad de desarrollar insights accionables a partir de datos complejos.

The "Introduction to Programming" course in the context of Business Analytics comprehensively addresses the fundamentals of programming, crucial for understanding how to develop and apply effective software solutions in business analysis. The program emphasizes the importance of basic programming principles and their direct application in business environments, ensuring that students acquire a solid foundation to handle various technical challenges.

The course modules cover a wide spectrum of essential topics. From studying types of data and basic operators, which are fundamental for handling data and executing logical and mathematical operations, to the manipulation and management of strings, and complex data structures, indispensable in handling large volumes of information in business.

Furthermore, techniques for using control instructions that direct the flow of program execution will be taught, crucial skills for logical problem solving in programming. The course also includes a deep dive into how to define and use functions and procedures to modularize the code, improve reusability, and simplify the structuring of complex programs.

Regarding conditional structures and loops, their use for performing repetitive tasks and making programmatic decisions will be explained, thereby increasing efficiency in solving specific business problems. In addition, advanced techniques for optimizing code efficiency and handling text files will be covered, skills necessary for data processing and analysis in various formats.

Finally, the course will highlight the applicability of these programming skills across a variety of industries, such as finance, retail, and technology. The relevance of programming in data analysis and business intelligence roles will be discussed, emphasizing how programming can significantly enhance the ability to develop actionable insights from complex data..

2.3. Contenido detallado

Presentación de la asignatura.
Explicación de la Guía Docente.

1. Módulo 1: Introducción a la programación
 - a. Fundamentos de la programación. Definiciones, historia y evolución de los lenguajes.
 - b. Aplicaciones en el mundo empresarial. Lenguajes de programación en Negocios. Herramientas de Desarrollo.
 - c. Inicio en la programación. Conceptos básicos y primer código.
2. Módulo 2: Tipos de Datos y Operadores en la Programación
 - a. Fundamentos de Programación.
 - b. Operadores en Programación.
 - c. Manejo y Conversión de Datos.
3. Módulo 3: Arrays, strings y estructuras
 - a. Arrays/Listas. Definición y uso. Operaciones Básicas. Iteración y Búsqueda.
 - b. Manipulación de Cadenas de Texto. Operaciones con cadenas. Búsqueda y sustitución. Formato.
4. Módulo 4: Instrucciones de control
 - a. Estructuras de decisión. If / Else.
 - b. Estructuras de Repetición. While / For.

[Examen Parcial]

5. Módulo 5: Funciones
 - a. Introducción a las Funciones en Programación. Concepto de Función. Ventajas de Usar Funciones.
 - b. Creación de Funciones Personalizadas. Sintaxis y Estructura. Parámetros y Argumentos. Devolución de Valores.
 - c. Funciones y su Importancia en la Automatización de Tareas.
6. Módulo 6: Condicionales
 - a. Conceptos Avanzados de Condicionales.
 - b. Condicionales Anidadas.
 - c. Operadores Lógicos en Condicionales
7. Módulo 7: Bucle
 - a. Bucles For y While en Profundidad.
 - b. Uso de Bucles en Procesamiento de Datos.
 - c. Ejemplos de Bucles en Aplicaciones de Negocios.
8. Módulo 8: Eficiencia, Ficheros de Texto y estructuras de programa
 - a. Conceptos de Eficiencia y Optimización de Código. Eficiencia. Optimización. No Optimizar Prematuramente. Identificación de Cuellos de Botella.
 - b. Ficheros de texto. Lectura y Escritura de Archivos Manipulación de Datos en Archivos.
 - c. Estructuras de Programa. Diseño y estructura de un programa. Función principal y específica. Punto de entrada y flujo de control.

2.4. Actividades Dirigidas

Durante el curso se podrán desarrollar actividades, prácticas, memorias o proyectos.

Al finalizar cada unidad didáctica, se podrá chequear los conocimientos relacionados con el temario así como solicitar la entrega de ejercicios teóricos y prácticos relacionados con el contenido del temario. Se podrán realizar prácticas de laboratorio evaluadas. El laboratorio dispondrá del material necesario (PC y programas) para el desarrollo de la actividad. También estará permitido el uso de un PC personal.

2.5. Actividades formativas

Actividades formativas:

Tipo de actividad modalidad presencial	Horas	Presencialidad %
A1 Clase Magistral/ Fundamentos Teóricos	35	100%
A2 Clases prácticas. Seminarios y talleres	25	100%
A3 Tutoría	9	100%
A4 Trabajos o ejercicios de los estudiantes	18	0%
A5 Actividades a través de recursos virtuales	0	0%
A6 Acceso e investigación sobre contenidos complementarios	6	0%
A7 Estudio individual	51	0%
A13 Evaluación	6	100%

Tipo de actividad modalidad a distancia	Horas	Presencialidad %
A9 Clases asíncronas	12	0%
A10 Clases prácticas. Síncronas o asíncronas	12	0%
A3 Tutoría	24	0%
A4 Trabajos o ejercicios de los estudiantes	18	0%
A5 Actividades a través de recursos virtuales	12	0%
A6 Acceso e investigación sobre contenidos complementarios	12	0%
A7 Estudio individual	54	0%
A13 Evaluación	6	100%

Metodologías docentes:

Presencial: MD1, MD2, MD3, MD4, MD5

A distancia: MD1, MD2, MD3, MD4, MD5

3. SISTEMA DE EVALUACIÓN

3.1. Sistema de calificaciones

El sistema de calificaciones finales se expresará numéricamente del siguiente modo:

- 0 - 4,9 Suspenso (SS)
- 5,0 - 6,9 Aprobado (AP)
- 7,0 - 8,9 Notable (NT)
- 9,0 - 10 Sobresaliente (SB)

La mención de "matrícula de honor" podrá ser otorgada a alumnos que hayan obtenido una calificación igual o superior a 9,0.

3.2. Criterios de evaluación

Convocatoria ordinaria

Modalidad: Presencial

Sistemas de evaluación	Porcentaje
S1 Asistencia y participación en clase	10%
S2 Presentación de trabajos y proyectos (Prácticas individuales y trabajo en equipo)	40%
S3 Prueba parcial presencial (escrita/presentación trabajo)	10%
S4 Examen final o trabajo final presencial	40%

Convocatoria extraordinaria

Modalidad: Presencial

Sistemas de evaluación	Porcentaje
S2 Presentación de trabajos y proyectos (Prácticas individuales y trabajo en equipo)	30%
S4 Examen final o trabajo final presencial	70%

Restricciones y explicación de la ponderación: Para poder hacer media con las ponderaciones anteriores será necesario obtener al menos una calificación de 5 en la prueba final.

Asimismo, será potestad del profesor solicitar y evaluar de nuevo las prácticas o trabajos escritos, si estos no han sido entregados en fecha, no han sido aprobados o se desea mejorar la nota obtenida en ambas convocatorias.

En todo caso, la superación de cualquier materia/asignatura está supeditada a aprobar las pruebas finales presenciales e individuales correspondientes.

3.3. Restricciones

Calificación mínima

Para poder hacer media con las ponderaciones anteriores es necesario obtener al menos una calificación de 5 en la prueba final.

Asistencia

El alumno que, injustificadamente, deje de asistir a más de un 25% de las clases presenciales

podrá verse privado del derecho a examinarse en la convocatoria ordinaria.

Normas de escritura

Se prestará especial atención en los trabajos, prácticas y proyectos escritos, así como en los exámenes tanto a la presentación como al contenido, cuidando los aspectos gramaticales y ortográficos. El no cumplimiento de los mínimos aceptables ocasionará que se resten puntos en dicho trabajo o examen.

3.4. Advertencia sobre plagio

La Universidad Antonio de Nebrija no tolerará en ningún caso el plagio o copia. Se considerará plagio la reproducción de párrafos a partir de textos de auditoría distinta a la del estudiante (Internet, libros, artículos, trabajos de compañeros...), cuando no se cite la fuente original de la que provienen. El uso de las citas no puede ser indiscriminado. El plagio es un delito.

En caso de detectarse este tipo de prácticas, se considerará Falta Grave y se podrá aplicar la sanción prevista en el Reglamento del Alumno.

4. BIBLIOGRAFÍA

Bibliografía básica

Hinojosa Gutiérrez, Á. (2015). Python paso a paso: (ed.). RA-MA Editorial.
Lee, K. D. (2011). Python programming fundamentals. London: Springer.
Martin, R. C. (2009). Clean code: a handbook of agile software craftsmanship. Pearson Education.

Bibliografía recomendada

Thomas, D. Pragmatic Programmer, The: From Journeyman to Master Andrew Hunt.
Fowler, M. (2018). Refactoring: improving the design of existing code. Addison-Wesley Professional.
Kleppmann, M. (2017). Designing data-intensive applications: The big ideas behind reliable, scalable, and maintainable systems. " O'Reilly Media, Inc."

Software recomendado

Thonny
Visual Studio Code
Notepad++