

PROGRAMA DE LA ACTIVIDAD

Introducción a Python para Ciencia de Datos

Curso virtual · 6 de noviembre al 5 de diciembre de 2026

1. Descripción de la actividad

Título	Introducción a Python para Ciencia de Datos
Tipo de actividad	Curso
Modalidad	Virtual, con encuentros sincrónicos por videollamada
Docente a cargo	Carlos Rubén Bageta
Fechas	Del viernes 6 de noviembre al sábado 5 de diciembre de 2026 (10 encuentros)
Días y horarios	Viernes de 19:30 a 21:30 h y sábados de 11:00 a 13:00 h
Carga horaria	20 horas de clases sincrónicas (10 encuentros de 2 horas) + 10 horas trabajo autónomo.
Destinatarios	Público general interesado en iniciarse en la programación y en la ciencia de datos
Requisitos previos	No se requieren conocimientos previos de programación. Se necesita una computadora con conexión a internet.

El curso introduce a las personas participantes en la programación con Python orientada al análisis de datos. Parte de la configuración del entorno de trabajo y de los fundamentos del lenguaje (tipos de datos, estructuras, funciones y estructuras de control) y avanza hacia el manejo de datos tabulares con la librería pandas: carga, exploración, selección, limpieza, agrupamiento y combinación de datos.

El aprendizaje se organiza en torno a cuatro miniproyectos, uno por cada bloque de contenidos, que integran lo trabajado en clase. Durante todo el cursado se incorporan herramientas de inteligencia artificial como apoyo al aprendizaje, bajo una política de uso responsable: la IA se usa para pedir pistas, entender errores o revisar el propio código, nunca para reemplazar el razonamiento de quien aprende.

2. Objetivos

Objetivo general

Que las personas participantes adquieran los fundamentos de la programación en Python y de la manipulación de datos tabulares con pandas necesarios para iniciarse en la ciencia de datos, y desarrollen criterio para utilizar herramientas de inteligencia artificial de manera responsable y transparente.

Objetivos específicos

Al finalizar el curso, las personas participantes serán capaces de:

- Configurar y utilizar un entorno de trabajo (terminal, Jupyter, Google Colab o VS Code) para escribir y ejecutar código Python.

- Emplear tipos de datos, variables, operadores, listas, tuplas, diccionarios, funciones y métodos para representar y procesar información.
- Aplicar estructuras de control (condicionales y bucles) para resolver problemas sencillos.
- Instalar y usar paquetes y reconocer las ventajas de numpy para el cálculo numérico.
- Cargar, explorar, seleccionar, filtrar, limpiar y transformar conjuntos de datos con pandas, y obtener resúmenes por grupos y combinar tablas.
- Integrar los conocimientos adquiridos en cuatro miniproyectos y explicar, probar y modificar el código propio.
- Utilizar herramientas de IA como apoyo al aprendizaje, declarando su uso y evaluando críticamente sus respuestas.

3. Contenidos

Los contenidos se organizan en cinco unidades, cada una desarrollada en dos encuentros consecutivos (viernes y sábado). Los temas indicados como optativos se ofrecen como lectura autónoma y no son necesarios para continuar con el curso.

Unidad 1. Entorno de trabajo y primeros pasos

- Presentación del curso: alcance, metodología y evaluación. Política de uso responsable de la IA.
- Introducción a la terminal: comandos básicos de navegación y gestión de archivos.
- Primeros pasos en Python: ejecución de sentencias simples, variables, print(), input() y scripts. Convenciones de nombre.
- Uso básico de Jupyter Lab, Google Colab y VS Code.
- **Miniproyecto:** primer programa integrador (Miniproyecto 1).
- **Optativo (lectura autónoma):** gestión de paquetes con pip y entornos virtuales.

Unidad 2. Tipos de datos, estructuras, funciones y métodos

- Tipos de datos: numéricos, booleanos y cadenas de texto. Conversión de tipos.
- Estructuras de datos: listas (slicing y modificación), tuplas, rangos y diccionarios. Conjuntos.
- Variables y operadores aritméticos, relacionales, lógicos y de asignación.
- Funciones incorporadas y funciones definidas por el usuario: argumentos, valores de retorno, alcance local y argumentos clave-valor.
- Métodos de objetos de uso frecuente.
- **Miniproyecto:** Sistema básico de gestión de estudiantes (Miniproyecto 2).
- **Optativo (lectura autónoma):** argumentos arbitrarios (*args y **kwargs) y funciones de librerías.

Unidad 3. Paquetes, numpy y estructuras de control

- Paquetes e instalación desde Jupyter Notebook. Paquetes relevantes en ciencia de datos.
- Introducción a numpy: arreglos y ventajas respecto de las listas.
- Estructuras condicionales (if, elif, else) y operadores lógicos (and, or, not).

- Bucles for y while. Control del flujo con break, continue y pass. Bloques anidados. Depuración de bucles y buenas prácticas.
- **Miniproyecto:** simulador básico de cajero automático (Miniproyecto 3).
- **Optativo (lectura autónoma):** comparación extensa entre listas y numpy (vectorización).

Unidad 4. Introducción a pandas

- Estructuras de datos principales: Series y DataFrames.
- Carga y almacenamiento de datos; el formato CSV como formato principal de trabajo.
- Exploración e inspección de datos.
- Indexación, selección y filtrado: loc, iloc y filtros booleanos.
- **Optativo (lectura autónoma):** lectura y escritura de archivos JSON y Excel; lectura avanzada de archivos (procesamiento por fragmentos y Parquet).

Unidad 5. Limpieza, agrupamiento y combinación de datos con pandas

- Limpieza y preparación de datos: valores faltantes, registros duplicados, transformación con map y apply, y renombrado de índices y columnas.
- Operaciones de grupo (groupby) y obtención de resúmenes por categoría.
- Combinación de conjuntos de datos: concat, merge y join.
- Cierre del curso: reflexión final sobre el uso de la IA.
- **Miniproyecto:** análisis básico de ventas con pandas (Miniproyecto 4).
- **Optativo (lectura autónoma):** discretización (binning) y creación de variables dummy.

4. Metodología

El curso se dicta en modalidad virtual, con encuentros sincrónicos por videollamada de 2 horas. Cada unidad cuenta con un notebook de Jupyter que funciona como guía de contenido: reúne la explicación de los temas, ejemplos de código ejecutables, ejercicios y el enunciado del miniproyecto correspondiente.

El trabajo se desarrolla de la siguiente manera:

- **Práctica guiada en clase.** Cada tema se presenta con ejemplos y se practica mediante ejercicios y puntos de control que se resuelven durante el encuentro. Estos ejercicios son parte de la práctica y no se entregan.
- **Miniproyectos.** Cada unidad culmina con un miniproyecto dividido en etapas obligatorias y opcionales. Cada uno incluye una declaración de uso de IA y una breve reflexión final.
- **Uso responsable de la IA.** Se permite consultar herramientas de IA para pedir pistas, interpretar errores, que expliquen código escrito por la propia persona, generar casos de prueba y comparar alternativas simples. No se permite pedirles que resuelvan el ejercicio o el miniproyecto, ni entregar código que no se pueda explicar. La regla de trabajo es: primero pienso, luego intento, después consulto a la IA, finalmente reviso, adapto y explico.
- **Contenidos optativos.** Los temas que profundizan o amplían el programa están señalados en los notebooks como lectura autónoma, para sostener un ritmo adecuado en los encuentros.

- **Entornos de trabajo.** Se utilizan Python con Jupyter Lab, Google Colab o VS Code, a elección de cada participante.

5. Cronograma

El curso comprende 10 encuentros, de dos por semana, entre el 6 de noviembre y el 5 de diciembre de 2026.

N.º	Fecha y horario	Temas	Actividades y producto
1	Vie 06/11 19:30 a 21:30 h	Unidad 1. Presentación del curso: alcance, metodología y evaluación. Política de uso responsable de la IA. Configuración del entorno. Terminal: navegación y gestión de archivos.	Práctica guiada en la terminal.
2	Sáb 07/11 11:00 a 13:00 h	Unidad 1. Python desde la terminal: cadenas, scripts, variables, print() e input(). Convenciones de nombre. Jupyter Lab, Google Colab y VS Code.	Actividad con IA: interpretar un error. Cierre de la unidad: Miniproyecto 1.
3	Vie 13/11 19:30 a 21:30 h	Unidad 2. Tipos de datos numéricos, booleanos y cadenas de texto. Conversión de tipos. Listas, tuplas, rangos y diccionarios.	Práctica guiada de lectura de código. Actividad con IA: autoevaluación de tipos de datos.
4	Sáb 14/11 11:00 a 13:00 h	Unidad 2. Operadores relacionales, lógicos y de asignación. Funciones incorporadas y definidas por el usuario. Métodos.	Actividades con IA: explicar código propio y generar casos de prueba. Cierre de la unidad: Miniproyecto 2.
5	Vie 20/11 19:30 a 21:30 h	Unidad 3. Paquetes e instalación desde Jupyter. Introducción a numpy. Condicionales (if, elif, else) y operadores lógicos.	Práctica guiada: corregir una condición mal planteada. Actividad con IA: pedir una pista.
6	Sáb 21/11 11:00 a 13:00 h	Unidad 3. Bucles for y while. Control del flujo (break, continue, pass). Bloques anidados. Buenas prácticas.	Práctica guiada: depurar bucles con errores. Actividad con IA: interpretar un error. Cierre de la unidad: Miniproyecto 3.
7	Vie 27/11 19:30 a 21:30 h	Unidad 4. Introducción a pandas: Series y DataFrames. Carga y almacenamiento de datos (CSV). Exploración e inspección de datos.	Práctica guiada con DataFrames pequeños.
8	Sáb 28/11 11:00 a 13:00 h	Unidad 4. Indexación, selección y filtrado de DataFrames (loc, iloc y filtros booleanos).	Práctica guiada de selección y filtrado. Actividad con IA: explicar código propio.
9	Vie 04/12 19:30 a 21:30 h	Unidad 5. Limpieza y preparación de datos: valores faltantes, duplicados, transformación con map y apply, renombrado de índices y columnas.	Práctica guiada: detectar nulos y duplicados. Actividad con IA: generar casos de prueba.
10	Sáb 05/12 11:00 a 13:00 h	Unidad 5. Operaciones de grupo (groupby). Combinación de datos: concat, merge y join. Cierre del curso.	Actividad con IA: comparar soluciones. Cierre de la unidad: Miniproyecto 4 y reflexión final sobre el uso de la IA.

Después del último encuentro. La entrega del documento con los miniproyectos y la instancia oral de evaluación (ver sección 6) se coordinan al finalizar el cursado; las fechas se comunicarán oportunamente a las personas inscriptas.

6. Evaluación y certificación

La evaluación se basa en los cuatro miniproyectos desarrollados durante el curso y en una instancia oral posterior. Los ejercicios de las clases son práctica guiada y no se entregan.

Instancias de evaluación

- **Entrega de los miniproyectos.** Cada participante presenta, en un único documento, los cuatro miniproyectos concluidos. Cada miniproyecto debe incluir su declaración de uso de IA y su reflexión final. Se aprueban cuando se cumplen las etapas obligatorias, de acuerdo con la rúbrica de evaluación.
- **Instancia oral.** Se realiza por videollamada. Consiste en tres preguntas sobre un miniproyecto que se asigna con anticipación, y apunta a verificar que la persona comprende, puede explicar y puede modificar el código que presentó. Se aprueba con al menos dos respuestas correctas.
- **Recuperatorio.** Quienes no aprueben la instancia oral pueden rendir un recuperatorio oral.

Certificación

Certificación	Condición
Certificado de aprobación	Haber entregado y aprobado los cuatro miniproyectos y haber aprobado la instancia oral.
Certificado de asistencia	Haber presentado los miniproyectos y no haber aprobado la instancia oral.
Sin certificación	No haber presentado los miniproyectos.

7. Bibliografía

McKinney, W. (2022). *Python for data analysis* (3.ª ed.). O'Reilly Media.

Python Software Foundation. (s. f.). *Python* (Versión 3) [Lenguaje de programación].
<https://www.python.org>

Shea, J. M. (2024). *Foundations of data science with Python* (1.ª ed.). CRC Press.