

March 25, 2020

# 1 Programación para *Data Science*

## 1.1 Unidad 5: Adquisición de datos en Python

En este Notebook encontraréis dos conjuntos de ejercicios: un primer conjunto de **ejercicios para practicar** y un segundo conjunto de **actividades evaluables** como PEC de la asignatura.

En cuanto al conjunto de ejercicios para practicar, éstos no puntúan para la PEC, pero os recomendamos que los intentéis resolver como parte del proceso de aprendizaje. Encontraréis ejemplos de posibles soluciones a los ejercicios al propio notebook, pero es importante que intentéis resolverlos vosotros antes de consultar las soluciones. Las soluciones os permitirán validar vuestras respuestas, así como ver alternativas de resolución de las actividades. También os animamos a preguntar cualquier duda que surja sobre la resolución de los **ejercicios para practicar** en el foro del aula.

Además, veréis que todas las actividades tienen una etiqueta que indica los recursos necesarios para llevarla a cabo. Hay tres posibles etiquetas:

- **NM Sólo materiales:** las herramientas necesarias para realizar la actividad se pueden encontrar en los materiales de la asignatura.
- **EG Consulta externa guiada:** la actividad puede requerir hacer uso de herramientas que no se encuentran en los materiales de la asignatura, pero el enunciado contiene indicaciones de dónde o cómo encontrar la información adicional necesaria para resolver la actividad.
- **EI Consulta externa independiente:** la actividad puede requerir hacer uso de herramientas que no se encuentran en los materiales de la asignatura, y el enunciado puede no incluir la descripción de dónde o cómo encontrar esta información adicional. Será necesario que el estudiante busque esta información utilizando los recursos que se han explicado en la asignatura.

Es importante notar que estas etiquetas no indican el nivel de dificultad del ejercicio, sino únicamente la necesidad de consulta de documentación externa para su resolución. Además, recordad que las **etiquetas son informativas**, pero podréis consultar referencias externas en cualquier momento (aunque no se indique explícitamente) o puede ser que podáis hacer una actividad sin consultar ningún tipo de documentación. Por ejemplo, para resolver una actividad que sólo requiera los materiales de la asignatura, podéis consultar referencias externas si queréis, ya sea tanto para ayudaros en la resolución como para ampliar el conocimiento!

En cuanto a la consulta de documentación externa en la resolución de los ejercicios, recordad **citar siempre la bibliografía utilizada** para resolver cada actividad.

---

## 1.2 Ejercicios y preguntas teóricas para la PEC

A continuación, encontraréis los **ejercicios y preguntas teóricas que debéis completar en esta PEC** y que forman parte de la evaluación de esta unidad.

### 1.3 Pregunta 1

La respuesta recibida después de realizar una petición a una web API http es un objeto que contiene, entre otros, los siguientes atributos: **status.code**, **content** y **headers**. Describe qué información contiene cada uno de los atributos anteriormente enumerados y pon un ejemplo de cada uno. Recordad que hay que citar las referencias consultadas para responder la pregunta, y que la respuesta que proporcionéis debe ser original (redactada por vosotros mismos, después de haber leído y entendido las referencias que consideréis oportunas). **(1 punto)** NM

**Respuesta**

### 1.4 Pregunta 2

Enumera tres librerías de Python para acceder a una API y especifica la API a la cual se accede. Para cada una de las librerías anteriormente enumeradas, pon un ejemplo de **endpoint** de la API que permite obtener una determinada información y de la función que permite obtenerla. Recordad que hay que citar las referencias consultadas para responder la pregunta. **(1 punto)** EG

**Nota.** Un ejemplo sería la librería de Python [Tweepy](#) que accede a la API de Twitter. Un posible endpoint sería `https://api.twitter.com/1.1/search/tweets.json` y la función `api.get_user()` permitiría obtener información de un determinado usuario.

**Respuesta**

### 1.5 Ejercicio 1

Implementad una función que devuelva una lista con el nombre de las personas actualmente en el espacio consultando alguna de las APIs que se detallan en la siguiente [url](#). **(1 punto)** NM

**Respuesta**

[ ]:

### 1.6 Ejercicio 2

Queremos saber el número de crímenes violentos que se han producido en Reino Unido en una localización (latitud, longitud) y fecha concretas mediante la siguiente [url](#). Implementad un conjunto de funciones para obtener el número de crímenes producidos en una determinada fecha en una determinada localización.

- La primera función devolverá la latitud y longitud de una determinada dirección postal mediante la API de geolocalización de [google maps](#). **(1 punto)**
- La segunda función devolverá el número de crímenes producidos en una determinada fecha y en una determinada localización (latitud, longitud). **(1 punto)**

Usa ambas funciones para obtener el número de crímenes violentos en la dirección **Adelaide St, WC2N 4HZ, London, United Kingdom** en abril del 2018.NM

**Nota:** Deberéis registraros a [Google Cloud Platform](#) para obtener las credenciales de la API de googlemaps.

**Respuesta**

[ ]:

### 1.7 Ejercicio 3

Queremos conocer los 25 nombres más frecuentes de los recién nacidos en Barcelona por sexo entre los años 1996 y 2018. Implementad una función de dos parámetros (id: identificador del recurso, año: año de consulta), que devuelva una lista con el nombre más frecuente para niña y para niño en dicho año. Usa la función para obtener lo siguiente:

- a) Una lista con los 25 nombres más frecuentes de los recién nacidos en Barcelona por sexo entre los años 1996 y 2018. **(1,5 puntos)**
- b) Crear una lista ordenada de mayor a menor, a partir de la lista obtenida en el apartado a), según el número de veces que dicho nombre ha sido utilizado. **(0,5 puntos)**

Para realizar el ejercicio consultad el portal de datos abiertos del Ayuntamiento de Barcelona mediante la siguiente [url](#). NM

**Nota 1:** Consultad como realizar las consultas mediante la API en [la pestaña de Desarrolladores](#)

**Nota 2:** Algunos nombres pueden contener espacios. Deberéis eliminar dicho espacio para realizar el conteo correctamente.

**Respuesta**

[ ]:

### 1.8 Ejercicio 4

[Billboard](#) es una revista semanal norteamericana sobre la industria musical. Programad un crawler que devuelva el nombre de los álbumes que se muestran en la lista [Year-End Billboard 200 Albums](#). Utilizad la estructura de crawler que hemos visto en el Notebook de teoría de esta unidad modificando únicamente dos líneas de código:

- URL de inicio.
- La expresión XPath que selecciona el contenido a capturar

**Nota:** si la ejecución del *crawler* os devuelve un error `ReactorNotRestartable`, reiniciad el núcleo del Notebook (en el menú: `Kernel - Restart`). **(2 puntos)** EG

**Respuesta**

[ ]:

## 1.9 Ejercicio 5

[Filmaffinity](#) es una web dedicada al cine para consultar toda la información relativa a las películas. Programad un crawler que devuelva solamente una tupla con las diez mejores películas y sus correspondientes directores. **(1 puntos)** EI

**Respuesta**

[ ]:

## 1.10 Ejercicio opcional

Programad una función que devuelva la fecha y hora de los 10 próximos pases de la estación espacial internacional ([ISS](#)) sobre la Torre Eiffel (especificada por su **longitud** y **latitud**). La función debe devolver una lista de 10 elementos, cada uno de los cuales debe ser una cadena de caracteres con la fecha y la hora de los pases.

**Respuesta**

[ ]: