



IIC2115 – Programación como Herramienta para la Ingeniería (I/2019)

Laboratorio 7 - Bases de datos

Objetivos

- Aplicar los contenidos de bases de datos para analizar y realizar consultas a conjuntos de información real.

Entrega

- **Lenguaje a utilizar:** Python 3.6
- **Lugar:** repositorio privado en GitHub. Recuerde incluir todo en una carpeta de nombre **L07**.
- **Entrega parcial:** lunes 6 de mayo a las 16:50 hrs.
- **Entrega final:** domingo 12 de mayo a las 23:59 hrs.
- **Formato de entrega:** archivo python notebook (**.ipynb**) con la solución, ubicado en la carpeta **L07**. Suba además, en la misma carpeta, un archivo **README.md** con las instrucciones para ejecutar su tarea. No se debe subir ningún otro archivo a la carpeta. Utilice múltiples celdas de texto y código para facilitar la revisión de su tarea. **No suba la base de datos o va a tener problemas al subir el laboratorio.**
- **Descuentos:** se descontará 0.5 pts. por cada hora de atraso y fracción en la entrega final. Tareas que no cumplan el formato de entrega tendrán un descuento de 0.5 pts.
- **Entregas parciales subidas fuera de plazo no serán consideradas.**
- **Tareas con errores de sintaxis y/o que generen excepciones serán calificadas con nota 1.0.**
- Las discusiones en las *issues* del Syllabus en GitHub son parte de este enunciado.

Introducción

Análisis del sistema Transantiago

Usted ha sido contratado por el Directorio de Transporte Público Metropolitano (DTPM) para analizar el rendimiento del transporte público. DTPM es el organismo público encargado de analizar de forma integral el sistema de transporte público capitalino y velar por la adecuada coordinación de los diferentes modos que participan en el transporte público de la ciudad de Santiago. Es por esto, que está muy interesado en saber cierta información respecto a los datos que se genera por medio de los GPS.

DTPM ha entregado a usted una base de datos con las emisiones reales de GPS de algunos recorridos de buses de Transantiago para una ventana de tiempo específica. Con esta información y la ayuda de SQLite, se espera que pueda resolver una serie de misiones descritas en la sección siguiente.

La base de datos

Los datos se encuentran disponibles en el siguiente [link](#), que descarga un archivo de texto plano de valores separados por coma (**csv** del inglés *comma-separated values*) llamado **emisiones.csv**. Cada fila del archivo posee valores separados por una coma (“,”). Los valores de la primera fila corresponden a los nombres de las columnas y el resto de las filas contienen la información asociada a emisiones GPS de buses. Cada fila corresponde a una emisión GPS y las columnas (o atributos) que posee una emisión se describen a continuación:

1. **measurement_id**: identificador interno de una medición GPS. Este valor corresponde a un número hexadecimal de largo fijo.
2. **expedition_id**: identificador interno de la expedición asociada a la medición GPS. Una expedición corresponde a la realización de un recorrido por un bus. Este valor corresponde a un número hexadecimal de largo fijo.
3. **dispatch_time**: instante de tiempo en que ocurre el despacho de la expedición asociada a la medición GPS. Este valor se encuentra representado en el formato **año-mes-día hora:minuto:segundo**, por ejemplo, “2018-04-01 14:33:21”.
4. **line_id**: identificador interno de la línea del bus. La línea de un bus se refiere a servicio prestado por el sistema. Este valor corresponde a un número hexadecimal de largo fijo.

5. **line_code**: código usuario de la línea. Este corresponde al código que llevan los buses en su parte frontal. Se encuentra represando por el código, por ejemplo “C02”.
6. **direction**: sentido de operación de la línea. Este campo indica si la operación es en el sentido “ida”, representado por una **I** o “regreso”, representado por una **R**.
7. **bus_id**: identificador interno del bus. Este valor corresponde a un número hexadecimal de largo fijo.
8. **license_plate**: placa patente del bus. Por ejemplo “BFRD-27” o “FG-3241”.
9. **bus_capacity**: capacidad física máxima del bus. Representado con un número que representa la cantidad de personas que caben en el bus.
10. **gps_time**: instante de tiempo en que ocurre la medición GPS. Este valor se encuentra representado en el formato **año-mes-dia hora:minuto:segundo**, por ejemplo, “2018-04-01 14:33:21”.
11. **latitude**: latitud de la medición GPS.
12. **longitude**: longitud de la medición GPS.
13. **distance_kms**: distancia recorrida por el bus desde el despacho hasta el instante de la emisión GPS en kilómetros.
14. **total_kms**: distancia total a recorrer por el bus en kilómetros.
15. **instant_speed**: velocidad instantánea al momento de registrar la medición GPS en km/h.
16. **measurement_speed**: estimación de la velocidad para el instante de la medición GPS, basada en la emisiones recientes en km/h.

Misiones (0.5 pts por misión correctamente ejecutada)

Para completar el requerimiento solicitado por DTPM, usted deberá realizar una serie de misiones. En la mayoría de ellas deberá responder una pregunta sobre los datos disponibles. Para hacerlo, construya consultas en lenguaje SQLite que le permita responder directamente la pregunta planteada. Para facilitar la corrección utilice la siguiente estructura de Python en cada una de las preguntas correspondientes:

```
def mision_x(params):  
    #Lógica y consultas SQL  
    return respuesta
```

Donde “x” es el número de la misión a responder y “params” (opcional) corresponden a los parámetros de la consulta si es que aplica. La respuesta debe poder visualizarse directamente en Python. A continuación se describen las misiones que deberá completar:

- M1. Para comenzar, es necesario tener nuestros datos cargados como una base de datos SQLite dentro de Python. Su primera misión será buscar la forma de importar un archivo de texto tipo **.csv** como una tabla de base de datos SQLite al código de Python. Recuerde que en SQLite también existen los tipos de datos, busque la forma de indicarle a SQLite a qué corresponde cada columna de datos (enteros, texto, decimales, fechas, etc.).
- M2. Ya tenemos los datos en una base de datos e integrados con python. Ha llegado la hora de realizar algunas consultas a la base de datos. Es importante saber con cuántos datos estamos trabajando. Realicé una única consulta que determine la cantidad de filas de datos la tabla y otra que muestre la cantidad de columnas.
- M3. ¿Qué expediciones se realizaron con una placa patente en concreto?. Identifique las expediciones mediante `expedition_id`, `line_code` y `direction`. **Ejemplo de input:** 'GHRD-15'.
- M4. ¿Qué emisiones GPS se realizaron por sobre cierta velocidad o superior?. Identifique las emisiones mediante `measurement_id`, `line_code`, `direction` y `gps_time`. **Ejemplo de input:** 15.
- M5. ¿Qué líneas de buses posee la base de datos? Retorne las líneas como `line_code` y `direction`
- M6. Cree una nueva columna en la base de datos que represente solo el día de la emisión en formato **año-mes-día**, ejemplo '2018-08-03'.
- M7. Cree una nueva columna en la base de datos que represente solo el día del despacho de cada una de las emisiones GPS. Esta debe estar en formato **año-mes-día**, ejemplo '2018-08-03'.
- M8. Obtenga todas las emisiones de un día cualquiera para un bus en específico. **Ejemplo de input:** '2018-08-03', 'GHRD-15'. **ejemplo de input:** '2018-08-10'.
- M9. ¿Qué buses son utilizados en más de una línea? ¿Cuántas veces es usado cada uno de ellos? Retorne las patentes de los buses asociados a la cantidad de líneas en que se usan (solo muestre los que se usen en más de una línea) Identifíquelos por la patente.
- M10. Retorne las emisiones GPS de la expedición con mayor número de emisiones para la línea sentido pedida. **ejemplo de input:** 'C02', 'I'.

- M11. ¿En qué día se utilizan la mayor cantidad de buses diferentes? Retorne una tabla con el o los días en que se utilice la mayor cantidad de buses diferentes asociado a esa cantidad.
- M12. Para cada servicio sentido (line_code, direction) determine el total de kilómetros-persona realizados en buses despachados entre las 13 pm y las 16 pm de cada día. Considere que los buses siempre están llenos en ese horario.

Corrección

Para la corrección de este laboratorio, se revisarán las consultas construidas para responder las diferentes preguntas propuestas. Además, se evaluarán las tablas definidas en la parte 2 y cómo son justificadas. Es decir, cómo diseñó su modelo de datos.

Avance parcial (final del la clase)

1. Responda las misiones que serán trabajadas en clases

Entrega final

1. Responda todas las misiones del enunciado

Política de Integridad Académica

“Como miembro de la comunidad de la Pontificia Universidad Católica de Chile me comprometo a respetar los principios y normativas que la rigen. Asimismo, prometo actuar con rectitud y honestidad en las relaciones con los demás integrantes de la comunidad y en la realización de todo trabajo, particularmente en aquellas actividades vinculadas a la docencia, el aprendizaje y la creación, difusión y transferencia del conocimiento. Además, velaré por la integridad de las personas y cuidaré los bienes de la Universidad.”

En particular, se espera que mantengan altos estándares de honestidad académica. Cualquier acto deshonesto o fraude académico está prohibido; los alumnos que incurran en este tipo de acciones se exponen a un procedimiento sumario. Ejemplos de actos deshonestos son la copia, el uso de material o equipos no permitidos en las evaluaciones, el plagio, o la falsificación de identidad, entre otros. Específicamente, para los cursos del Departamento de Ciencia de la Computación, rige obligatoriamente la siguiente política de integridad académica en relación a copia y plagio: Todo trabajo presentado por un alumno (grupo) para los efectos de la evaluación de un curso debe ser hecho individualmente por el alumno (grupo), sin apoyo en material de terceros. Si un alumno (grupo) copia un trabajo, se le calificará con nota 1.0 en dicha evaluación y dependiendo de la gravedad de sus acciones podrá tener un 1.0 en todo ese ítem de evaluaciones o un 1.1 en el curso. Además, los antecedentes serán enviados a la Dirección de Docencia de la Escuela de

Ingeniería para evaluar posteriores sanciones en conjunto con la Universidad, las que pueden incluir un procedimiento sumario. Por “copia” o “plagio” se entiende incluir en el trabajo presentado como propio, partes desarrolladas por otra persona. Está permitido usar material disponible públicamente, por ejemplo, libros o contenidos tomados de Internet, siempre y cuando se incluya la cita correspondiente.