



Universidad del Desarrollo
Universidad de Excelencia

Taller de Programación

Certamen 2

14 de Junio de 2020

Instrucciones Generales

Resuelva con un compañero/a, la siguiente lista de ejercicios. Regístrese en uno de los grupos correspondientes a la evaluación Tarea2 disponibles en la sección Personas de canvas. Su código debe estar contenido en un único archivo `t2_apellido1_apellido2.py`. Este y los archivos correspondientes deben ser entregados en un archivo comprimido `.zip` con el formato `t2_apellido1_apellido2.zip` en canvas.

Detalles de la Entrega

- La fecha de entrega es el 28 de Junio del 2020 a las 23:59 hrs.
- Si su código tiene errores de sintaxis (no se ejecuta), su tarea no se considerará entregada.
- Sus soluciones pueden incluir todas funciones adicionales que usted estime necesarias
- Si utiliza métodos de librerías no vistas en clases, comente brevemente su funcionalidad y agregue un link a la documentación oficial.
- Recuerde que solo puede comentar sus soluciones con su profesor, los ayudantes y su compañero(a) de equipo.
- Su solución, junto con las soluciones de todas las secciones, será evaluada por un software detector de plagio. De detectarse copia, su certamen será evaluado con la **NOTA MÍNIMA** y la situación será informada al comité de ética de la Facultad, lo que puede derivar en una causal de eliminación de la universidad.

Contexto

Ustedes han sido contratados como analistas de datos para una institución que está realizando un seguimiento de la epidemia de covid-19 en el país y en el mundo, por lo que se les solicita programar algunas funciones y elaborar indicadores específicos para el estudio de la pandemia.

Archivos

Actualmente existen varios repositorios de datos que mantienen información actualizada sobre la epidemia de covid-19. El archivo `WHO-COVID-19-global-data.csv`, proveniente del sitio <https://covid19.who.int/> contiene la evolución del número de contagiados de covid-19 por país.

En este archivo existen los siguientes campos:

- `date_reported` : fecha para la cual se reportan los datos en el formato AAAA-MM-DD.
- `country_code` : código del país al que corresponden los datos.
- `country` : nombre del país al que corresponden los datos.
- `who_region` : región del planeta a la que pertenece el país de acuerdo a la OMS.
- `new_cases` : número de casos nuevos para la fecha del registro.
- `cumulative_cases` : número acumulado de casos a la fecha del registro.
- `new_deaths` : número de muertes nuevas para la fecha del registro.
- `cumulative_deaths` : número acumulado de muertes a la fecha del registro.

Por otro lado, el repositorio oficial de datos del Ministerio de Ciencia de Chile <https://github.com/MinCiencia/Datos-COVID19/>, mantiene los casos confirmados de covid-19 a nivel de comuna, disponibles en el archivo `Covid-19.csv`, el número de pacientes en la UCI a nivel de región disponibles `UCI_T.csv` y el índice de movilidad (cuanto se mueven las personas) interno y externo para todas las comunas del país, en los archivos `IndiceDeMovilidad-IM_externo.csv` y `IndiceDeMovilidad-IM_interno.csv` respectivamente. Los detalles de los campos de cada archivo pueden ser revisadas en el repositorio.

Con estos archivos y los aprendido en clases, resuelva los problemas descritos en la sección **Problemas**.

Problemas

1. Casos por país [0.5 pts.] Programe la función `casos_pais(file, country, fecha)` que recibe un archivo del tipo `WHO-COVID-19-global-data.csv`, el nombre de un país y una fecha en el formato `AAAA-MM-DD` y retorna una tupla con el país, la fecha y el respectivo número de contagiados de covid-19.

2. Ranking por país [0.5 pts.] Programe la función `n_caos_pais(n, fecha, file)` que retorna los `n` países con más contagiados para una fecha específica en el formato `AAAA-MM-DD` y un archivo del tipo `WHO-COVID-19-global-data.csv`.

3. Máximo de fallecidos [0.5 pts.] Programe la función `max_fallecidos(file)` que retorna una tupla con el país con más fallecidos y el número de estos.

4. Comparación de fallecidos [0.5 pts.] Programe la función `compara_fallecidos_paises(pais1, pais2, fecha, file)` que recibe el nombre de dos países, una fecha específica en el formato `AAAA-MM-DD` y un archivo de tipo `WHO-COVID-19-global-data.csv` y retorna el país que tiene el mayor número de fallecidos por covid-19 para esa fecha, y el país con el mayor número de fallecidos acumulados a la fecha.

5. Casos por región [0.5 pts.] Programe la función `contagiados_region(file)` que recibe un archivo del tipo `Covid-19.csv` y retorna un diccionario con el número de personas contagiadas con covid-19 por región. Las llaves del diccionario deben ser los nombres de la región y los valores el número de casos.

6. Casos por región y comuna [0.5 pts.] Programe la función `contagiados_file_r(file, region)` que reciba un archivo del tipo `Covid-19.csv` y el nombre de una región y retorna un archivo `region.csv` con los casos confirmados de covid-19 por comuna para una región específica. Utilice como nombre del archivo el nombre de la region en minúsculas, sin acentos y reemplazando los espacios por `_`. Por ejemplo, el archivo con datos para la región Arica y Parinacota debe llamarse `arica_y_parinacota.csv`.

7. Internados por región [0.5 pts.] Programe la función `internados_uci_region(file)` que reciba como entrada el archivo `UCI_T.csv` y retorne un diccionario con el último número de internados en UCI por región. El último número corresponde al registro más reciente (última fila). Las llaves del diccionario deben ser los nombres de la región y los valores el número de internados.

8. Evolución de pacientes UCI en el tiempo [1.0 pts.] Programe la función `porcentaje_poblacion(file, region)` que reciba el archivo `UCI_T.csv` y el nombre de una región, y retorna la fecha y el porcentaje de la población de esa región que está internada en la UCI. Grafique la evolución de este porcentaje en el tiempo para una región a elección (puede utilizar matplotlib o excel para realizar este gráfico) y guarde el gráfico en un archivo `.pdf` o `.png`. Comente sus observaciones respecto del crecimiento o decrecimiento del número de internados con respecto al tiempo en el archivo. No olvide agregar el título y el nombre de los ejes correspondientes al gráfico.

9. Movilidad por comuna [0.5 pts.] Programe las funciones `mov_comuna(file, comuna, fecha)` y `mov_comuna(file, comuna, fecha)_e` que reciben archivos del tipo `IndiceDeMovilidad-IM_interno.csv` y `IndiceDeMovilidad-IM_externo.csv` y que retornan el índice de movilidad interno y externo respectivamente, para una comuna y una fecha específica.

10. Cambio de movilidad [1.0 pts.] Programe la función `movi_comuna_file(file1, file2, comuna)` que recibe los archivos `IndiceDeMovilidad-IM_interno.csv` y `IndiceDeMovilidad-IM_externo.csv` y crea un único archivo `movilidad_comuna.csv` con la evolución de la movilidad de la comuna en el tiempo donde la primera columna contiene la fecha, la segunda columna la movilidad interna y la tercera columna la movilidad externa. Grafique la evolución de la movilidad interna y externa de dos comunas a elección (puede utilizar matplotlib o excel para realizar este gráfico) y guarde el gráfico en un archivo `.pdf` o `.png`. Comente sus observaciones respecto de la evolución de la movilidad para las comunas elegidas en el archivo. No olvide agregar el título, el nombre de los ejes correspondientes al gráfico y una leyenda.

NOTA Todas las funciones deben solicitar al ejecutarse si guardan el resultado un en archivo, de responder "SI" el programa debe guardar los resultados en un archivo agregando la cabecera(nombre de las columnas)