

Prueba - Ruby

Instrucciones:

- Leer la prueba completa antes de comenzar.
- Concentrarse en incluir la mayor cantidad de items funcionales posibles.
- Se debe utilizar **GIT** para el control de versiones.
- La prueba debe ser publicada en un repositorio único en GitHub, independiente de cualquier otro proyecto realizado anteriormente.
- Se deberá subir a la plataforma de [empieza](#) el link directo al repositorio.
- La prueba debe ser realizada y entregada antes de finalizar la clase (17.00hrs).
No se revisarán commits posteriores a la hora de entrega.
- Se puede utilizar material de apoyo como *Google, StackOverflow, Blogs*, material previo del curso, entre otros.
- El puntaje total de la prueba es de 100 puntos.
- Para aprobar, se debe obtener sobre un 70%.
- Copia de código fuente será castigada con nota 0.

Contexto

Se tiene un archivo con formato `.csv` que contiene nombres de alumnos con sus respectivas notas.

El archivo tiene la siguiente estructura:

```
David, 9, 2, 5, 1, 1
Macarena, 10, 2, A, 8, 1
Ignacio, 10, 10, 9, 9, A
Rocio, 10, 10, 10, 10, 10
```

Donde la A señala que el alumno estuvo ausente en la prueba.

Crea un nuevo archivo *alumnos.csv* y pega el listado de alumnos y sus notas.

Para leer el contenido de este archivo utilizaremos el siguiente método:

```
def read_alum(file_name)
  file = File.open(file_name, 'r')
  alum = file.readlines.map(&:chomp).map { |lines| lines.split(',') }
  file.close
  alum
end
```

Este método recibe como argumento el nombre del archivo y retorna un array con su contenido.

Requerimientos:

Crear un menú con 4 opciones con las siguientes funcionalidades:

- **Opción 1:** Imprimir en pantalla el nombre de cada alumno y el promedio de sus notas (30 puntos)

Utilizar el método de lectura facilitado.

- **Opción 2:** Imprimir en pantalla el nombre de cada alumno y la cantidad de inasistencias (30 puntos)

- **Opción 3:** Imprimir en pantalla los nombres de los alumnos aprobados (30 puntos)

Crear un método que reciba -como argumento- la nota necesaria para aprobar. Por defecto esa nota debe ser 5.0

- **Opción 4:** Termina el programa (5 puntos)
- Se debe validar que la opción escogida sea 1, 2, 3 ó 4. (5 puntos)

Si se ingresa cualquier otra opción el programa debe:

- Mostrar que la opción es inválida
- Mostrar nuevamente el menú
- Dar la posibilidad de que el usuario vuelva a ingresar una opción.