

Ensayo examen

Se pide desarrollar una aplicación para la gestión de partidos de un torneo de videojuegos. De cada partido se guarda la siguiente información:

- `id_partido`: identificador del partido (str)
- `cod_jugador`: código del jugador que participó (str)
- `puntos`: lista de enteros con las puntuaciones obtenidas en cada ronda (int).

Ejemplo del archivo `partidos.csv`:

```
M001:J101:100:200:150
M002:J102:300:250
M003:J101:400:350:300
M004:J103:200
M005:J102:150:250
```

Tras cargar los datos, se espera el siguiente diccionario por cada jugador.:

```
{
    "id_partido": "M001",
    "cod_jugador": "J101",
    "puntos": [100, 200, 150]
}
```

1. Ejercicio 1

Crea una función `cargar_partidos` que reciba una lista vacía y cargue los datos desde el archivo `partidos.csv` en el formato especificado.

```
partidos = []
cargar_partidos(partidos)
print(len(partidos)) # Debería devolver 5
print(partidos[0])   # {'id_partido': 'M001', 'cod_jugador':
                      'J101', 'puntos': [100, 200, 150]}
```

2. Ejercicio 2

Escribe una función `promedio_puntos` que calcule el promedio de puntos por ronda en todos los partidos.

```
print(promedio_puntos(partidos)) # Debería devolver 233.3
```

3. Ejercicio 3

Escribe una función `calcular_puntos_totales` que reciba el código de un jugador y la lista de partidos, y devuelva el total de puntos obtenidos por ese jugador.

```
print(calcular_puntos_totales('J101', partidos)) # Debería devolver 1500
print(calcular_puntos_totales('J102', partidos)) # Debería devolver 950
```

4. Ejercicio 4

Crea una función `partidos_por_jugador` que devuelva un diccionario con el número de partidos jugados por cada jugador.

```
partidos_jugador = partidos_por_jugador(partidos)
print(partidos_jugador) # {'J101': 2, 'J102': 2, 'J103': 1}
```

5. Ejercicio 5

Desarrolla la función `jugador_mas_activo` que reciba el diccionario del ejercicio anterior y devuelva el código del jugador con más partidos jugados.

```
print(jugador_mas_activo(partidos_jugador)) # Debería devolver 'J101'
```

6. Ejercicio 6

Implementa una función `guardar_partidos_jugador` que reciba la lista de partidos y el código de un jugador, y guarde en un archivo `partidos_<cod_jugador>.csv` los datos de sus partidos (id, total de puntos, puntuaciones por ronda).

```
guardar_partidos_jugador(partidos, 'J101')
```

Archivo resultante `partidos_J101.csv`:

```
M001;450;100;200;150
```

M003;1050;400;350;300