

Grupo	P1 (1.5)	P2 (3.0)	P3 (1.5)	Comentarios
1	1.5	3	1.5	Init de Match tiene *args y **kwargs pero no se usan.
2	1.5	1.5	1.3	Codigo no compila. Los métodos load_matches y get_standings son demasiado largos. Es buena práctica inyectar las dependencias y no crearlas dentro de la misma clase (como lo que hicieron con el objeto db de la clase league). El método print_standings tiene demasiada responsabilidad (consulta la db, ordena e imprime). Hay varios metodos con mucha responsabilidad.
3	1.5	2.25	1.5	Es buena práctica inyectar dependencias y no crearlas dentro de la misma clase (parser = LeagueParser(path) dentro de la clase League). La funcion print_standings ademas de imprimir las posiciones calcula los puntos, esto puede ser separado.
4	1.5	2.8	1.5	La variable team_separator siempre tiene el mismo valor y solo se usa en una función, por lo que no hace mucho sentido que sea una variable (por ej podría ser un parámetro por default del método parse_game_line). Es buena práctica inyectar dependencias y no crearlas dentro de la misma clase (FileReader()) dentro de League). El método print_standings además de imprimir, ordena los datos (tratar que eso no pase ya que es poco cohesivo).
5	1.5	2.8	1.5	La función print_standings ademas de imprimir , ordena (recordar primer principio Solid).
6	1.5	2.25	1.3	Es buena práctica inyectar dependencias y no crearlas dentro de la misma clase (MatchParser). Es buena práctica no usar directamente los atributos de un objeto si no que acceder a ellos mediante funciones (Team.score). Si uno llama la función assign_points de Match más de una vez se asignan más de una vez los puntos, tratar de evitar eso.
7	1	2.8	1	La función last_function tiene un nombre poco descriptivo, además, ordena e imprime (poco cohesivo). Mezcla de español e inglés en algunas partes del código. Se menciona un code smell que no es de la categoria Bloaters.
8	1.5	2.05	1.3	El método print_standings además de imprimir, ordena los datos (tratar que eso no pase ya que es poco cohesivo). Es buena práctica inyectar dependencias y no crearlas dentro de la misma clase. La lectura de los Matches se hace cuando se llama la funcion print_standings, debería hacerse antes. La función get_winner hace más sentido en la clase Match. La representación usada en la línea 98 corresponde a un Code Smell, pudieron haber usado alguna estructura para guardar esa información.
9	1.5	2.05	1.3	No se debe leer el archivo al llamar la función para imprimir. La función print_standings tiene mucha responsabilidad (llama al lector de archivos, calcula puntos, ordena e imprime). Los equipos con su score se podrían guardar en una estructura y no en un array.
10	1.5	2.25	1.5	El atributo standings se repite en el init de la clase League. La función print_standings ademas de imprimir hace un orden de las posiciones. En vez de manipular los teams como un hash y despues convertir a arreglo de tuplas, se podría manejar como una estructura.
11	1.5	2.05	1.5	La clase League todavía es muy grande y con mucha responsabilidad. La funcion print_standings no solo imprime si no que abre el archivo, obtiene los teams y los partidos, obtiene los puntos, ordena los teams según el score y después imprime. Se podría usar una estructura para almacenar un equipo y sus puntos.
12	1.5	2.05	1.5	El método print_standings tiene mucha responsabilidad (consulta el archivo, obtiene los puntos de cada equipo, ordena e imprime). Es buena practica usar inyección de dependencias. La herencia FileParser-LeagueParser no hace mucho sentido considerando que el método find_team_indexes de la clase padre está muy acoplado al formato del archivo de la clase hija (podría ser una sola clase o agregar ese método a la clase hija), recordar que las herencias se usan para reutilizar código y aquí no tiene sentido eso. La clase League podría tener un atributo para almacenar los partidos y los equipos en vez de que solo sean variables locales de la función print_standings. Podrían haber almacenado el score del team en la misma clase team en vez de crear una tupla o haberlo almacenado con una estructura adecuada.
13	1.5	0	0.3	El código no tiene modificaciones.
14	1.5	2.05	1.5	La funcion print_standings no solo imprime si no que abre el archivo, obtiene los teams y los partidos, obtiene los puntos, ordena los teams según el score y después imprime, debería haber una función que haga exclusivamente la lectura del archivo y que ordene los teams (que no se llame en print_standings).
15	1	2.05	1	El code smell Long Parameter List no está presente en este código, la descripción que das corresponde al code smell Data Clump. El cálculo de scores y el orden de las posiciones debería hacerse en una función distinta a print_standings. No es intuitivo que la clase Team tenga un atributo teams.
16	1.5	2.25	1.5	La funcion print_standings no solo imprime si no que abre el archivo, obtiene los teams y los partidos, obtiene los puntos, ordena los teams según el score y después imprime, debería haber una función que haga exclusivamente la lectura del archivo y que ordene los teams (que no se llame en print_standings)
17	1.5	2.25	1.5	La funcion print_standings no solo imprime si no que abre el archivo, obtiene los teams y los partidos, obtiene los puntos, ordena los teams según el score y después imprime, debería haber una función que haga exclusivamente la lectura del archivo y que ordene los teams (que no se llame en print_standings). Se podría guardar el team y el puntaje en una estructura más semántica que la tupla.
18	1.5	2.8	1.5	La función print_standings no debe ordenar los datos.
19	1.5	0	0.3	Código no hace nada
20	1.5	2.8	1.3	La funcion print_standings no solo imprime si no que abre el archivo, obtiene los teams y los partidos, obtiene los puntos, ordena los teams según el score y después imprime, debería haber una función que haga exclusivamente la lectura del archivo y que ordene los teams (que no se llame en print_standings).
21	1.5	2.8	1.5	La funcion print_standings no solo imprime si no que abre el archivo, obtiene los teams y los partidos, obtiene los puntos, ordena los teams según el score y después imprime, debería haber una función que haga exclusivamente la lectura del archivo y que ordene los teams (que no se llame en print_standings). El método parse_league es muy largo.