

Sistema de controle da Série A!

Flávio Motta
Paradigma de Linguagem de Programação
UNIFAMINAS

2021-1

Descrição

”Bola na área pra ninguém cabecear”, esse não é nosso lema. A CBF perdeu o sistema que gerenciava o campeonato Brasileiro de Futebol e veio a nossa empresa pedir ajuda. Nossa equipe é reduzida (2 alunos) e temos muito código para implementar, então não podemos perder tempo. Fique atento pois nosso prazo não é longo, e logo devemos entregar um sistema de perfeita execução e estabilidade para a dona CBF.



Entrega

O sistema deverá ser entregue por email para flavioaam@hotmail.com no dia 13/04/2021 com horário limite as 23:59. Todos os arquivos do sistema deverão estar no formato .zip ou .rar e deverão conter os nomes dos integrantes. Não serão aceitos trabalhos entregues após o horário limite. A segunda entrega (entrega final) deverá ser feita no dia 22/06/2021 com horário limite as 23:59. Todos os arquivos do sistema deverão estar no formato .zip ou .rar e deverão conter os nomes dos integrantes.

Classes do sistema

Todas as classes devem ter implementação do método construtor e dos gets e sets necessários para uso do sistema.

1. Classe principal

- Esta classe será apenas para gerenciar o campeonato, dela que criaremos o campeonato e definiremos todos parâmetros necessários.
- Esta classe consta nos arquivos da turma, basta seguir as instruções na mesma

2. Pessoa

- Atributos
 - (a) String Nome
 - (b) int CPF
 - (c) Calendar Nascimento
 - (d) *Mais dois atributos a sua escolha, tem de ser atributos que façam sentido*
- Métodos
 - (a) int CalculaIdade()
 - Este método deve pegar a data de nascimento e retornar um int correspondendo a idade da pessoa
 - (b) *Método a sua escolha, tem que ser método que faça sentido*

3. Jogador (Essa classe deve herdar de Pessoa)

- Atributos
 - (a) Time atual
 - (b) Double salario

4. Time

- Atributos
 - (a) String Nome
 - (b) String cidade
 - (c) Jogador[] Elenco
- Métodos
 - (a) int contrataJogador(Jogador comprado, double salario)
 - Este método deve realizar a compra de um jogador de um clube para o atual.

5. Partida

- Atributos

- (a) Time mandante
- (b) Time visitante
- (c) Jogador[] titularA
- (d) Jogador[] titularB
- (e) int golsA
- (f) int golsB
- Métodos
 - (a) String getCidadeJogo()
 - Este método deve retornar a cidade do jogo baseada no time mandante.
 - (b) int getResultado()
 - Este método deve retornar -1 quando o time visitante foi vencedor, 0 quando der empate e 1 quando o mandante venceu.
 - (c) int printResultadoRegistrado()
 - Este método deve imprimir o placar do jogo no seguinte formato "nome-Mandante golsMandante x golsVisitante nomeVisitante"

6. Tabela

- Atributos
 - (a) String classificacao

7. Rodada

- Atributos
 - (a) Partida[] jogos

8. Campeonato (Este arquivo consta nos arquivos da turma, já tem os métodos pedidos porém sem implementação ainda).

- Atributos
 - (a) Time[] SerieA
 - (b) int ano
 - (c) Tabela tabela
 - (d) Rodada[] rodadas
- Métodos
 - (a) addTime(Time novo)
 - Método que adiciona o time passado por parâmetro ao vetor de times SerieA, lembre-se que o campeonato deve ter apenas 6 times!
 - (b) printTimes()
 - Método para imprimir todos os times cadastrados até então
 - (c) novaRodada(int novaRodada)

- Método que permite a criação e edição de uma nova rodada, ao selecionar este método lembre-se de imprimir o estado atual da rodada e permitir que o usuário escolha qual partida da rodada deseja editar. Lembre-se que existe um limite de rodadas no campeonato!
- (d) `checaFraudes()`
 - Método que verifica se há CPFs duplicados nos cadastros de todos jogadores do campeonato, se tiver, imprimir os nomes dos jogadores.
- (e) `tranferencia()`
 - Método para realizar a tranferencia de um jogador de um time para outro.
- (f) `listaCidades()`
 - Método para imprimir as cidades que estão presentes no campeonato, lembre-se que se uma cidade aparece mais de uma vez ela deve ser mostrada apenas uma vez.
- (g) `classificacao()`
 - Método que imprime a classificação atual do campeonato (pode mostrar apenas os times e sua pontuação atual)