You are the general manager of a football team (NFL).

As a general manager, it is your job to put together the best roster possible to win a championship against the other teams in the league.

League

Has a list of teams

Team

Has a roster

Points field (statistic)

Points Allowed field (statistic)

Touchdowns field (statistic)

Interceptions field (statistic)

Sacks field (statistic)

Use properties to access fields

Roster

Has a list of players

Average overall rating of players

Average age of players

Average weight of players

Player

Name field

Age field

Weight field

Position field (what position the player plays)

Overall rating field

Perform action method that will be overriden on each child class (For example, quarterback passes, running back rushes, etc.)

Enum for if the player plays offense, defense, and special teams

Position (quarterback, running back, wide receiver, etc.) inherit from player class

Quarterback:

Passing percentage field

Passer rating field

Passing touchdowns field

Running back:

Rushing attempts field

Rushing yards field

Rushing touchdowns field

Wide Receiver:

Receptions field

Receiving yards field

Receiving touchdowns field

Use properties to access fields

Other Requirements:

Overload plus (+) operator to acquire a new player to your team via trade.

Overload minus (-) operator to trade one of your players to a new team.

Use custom iterator to get the top five players from any team based on overall rating

Read/Write your roster to file

\* Don't get hung up on the stats. You can make them up. Although, it would be cooler if they were accurate.

\* Take the position classes as far as you want, but I would like to see a quarterback, running back, wide receiver

\* These classes will likely need more fields and methods than what it is provided in the spec