

Common SW candidate test

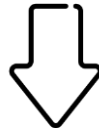
Create Console application that will do following:

1. Print on console median of Completed property from items provided by `ItemRepository.GetAllItems()` method from `CommonSwCandidateTest.Data.dll`
2. Create method which will create hierarchical tree of items provided by `ItemRepository.GetAllItems()` method from `CommonSwCandidateTest.Data.dll`. Items on same level will be sort by `item.PlannedStart`, `item.PlannedEnd`, `item.Name` properties. Print this tree to console – print properties `Id`, `Parent Id`, `Name`, `Planned Start` and `Planned End`.
3. Create method which will recalculate completed property. If item has some child items, it's Completed property is calculated from it's childs. If child item Completed property is also calculated, then use this calculated value to compute parent item completion.

Example:

Item 1 (completed 10%)

- Item 2 (completed 30%)
 - o Item 3 (completed 50%)
 - o Item 4 (completed 70%)
- Item 5 (completed 20%)



will be

Item 1 (completed = $(20+60)/200 = 40\%$)

- Item 2 (completed = $(50+70)/(200) = 60\%$)
 - o Item 3 (completed 50)
 - o Item 4 (completed 70)
- Item 5 (completed 20%)

4. Save hierarchical tree of items with recalculated "Completed" property to `C:\dev\hiararchical-{your-name}.csv`. One item = one line in csv. Hierarchical tree of items will contain following properties:
 - Level – root items have level 1, their direct subitems have level 2, etc.
 - Name
 - PlannedStart
 - PlannedEnd
 - Completed – if item has no sub items, then this property will be displayed. If item has sub items, Completion is recalculated as described above.
 - Was completion calculated – true/false