This file was generated with Violet UML Editor 2.1.0.   (View Source / Download Violet)

BTSNode<E>

+data:E
+idx: Integer
+left: SearchTreeNode<E>
+right: SearchTreeNode<E>

+SearchTreeNode(data:E,idx: integer)
+SearchTreeNode(data:E,idx:integer,
left:SearchTreeNode<E>,
right: SearchTreeNode<E>)

BST<E>

−overallRoot: SearchTreeNode

+BinarySearchTree():
+add(value :E, idx:integer);
− add(root:SearchTreeNode<E>,
 value:E,iidx:integer): SearchTreeNode<E>

+getSize(): Integer
−getSize(root:BSTNode<E>): Integer
+getList(size:integer):int[]
−getList(root:BSTNode<E> , arr: int[] , index:Integer):integer
+searchRange(start:E, end:E):int[]
−searchRange(root:BSTNode<E>,start: E, end:E,
arrList:E,ArrayList<Integer>):ArrayList<Integer>
−balanceTree(arr:arr:ArrayList<BSTNode<E>> ,start:integer, end: integer):BSTNode
+balanceTree()
−getNodeList(root:BSTNode<E>, arr:ArrayList<BSTNode<E>>, index:integer):integer
−getNodeList():BSTNode<E>
−getHeight(root:BSTNode<E>):integer
−isBalaced(root:BSTNode<E>):boolean
+isBalanced():booealn

Wearable

−ranking: Integer
−name: String
−price: Double
−bodyLocation: String
−category: String
−companyName: String
−url: String
−mapLocation: String
−city: String
−state: String
−country: String

+Wearable(ranking: Integer,name: String,price: Double,
bodyLocation: String,category: String, companyName: String,
url: String,mapLocation: String,city: String,state: String,
country: String)

+setName(name:String)
+getRanking(): Integer
+getName(): String
+getPrice(): Double
+getBodyLocation(): String
+getCategory(): String
+getCompanyName(): String
+getUrl(): String
+getMapLocation(): String
+getCity(): String
+getState: String
+getCountry: String

+toCSVFormat():String
+toString(): String

SearchTree<E>

−overallRoot: SearchTreeNode

+BinarySearchTree():
+add(value :E, idx:integer);
− add(root:SearchTreeNode<E>, value:E,
idx:integer): SearchTreeNode<E>
+getSize(): Integer
−getSize(root:BSTNode<E>): Integer
+getList(size:integer):int[]
−getList(root:BSTNode<E> , arr: int[] , index:Integer):integer
+searchRange(start:E, end:E):int[]
−searchRange(root:BSTNode<E>,start: E, end:E,
arrList:ArrayList<Integer>):ArrayList<Integer>

**<<Interface>>
Comparable**

Wearables

−wearableList: Wearable []
−rankingTree: BinarySearchTree<E>
−companyNameTree:SearchTree <E>
−priceTree: SearchTree <E>

+Wearables()
+readFile(fileName:String)
+getWearableAtIndex(index:integer):Wearable
+getRankingPositionData(): int[]
+getPricePositionData():int[]
+getCoNamePositionData():int[]
+generateCsv(positions): int[], filename: String):boolean
+searchRange(start:String, end:String):int[]
+searchRange(start:Integer, end:integer):int[]
+searchRange(start:double, end:double):int[]
+balanceRankingTree()
+toString(): String

SearchTreeNode<E>

+data:E
+idx: Integer
+ down: DupNode
+left: SearchTreeNode<E>
+right: SearchTreeNode<E>

+SearchTreeNode(data:E,iidx: integer)
+SearchTreeNode(data:E,idx:integer,
down:SearchTreeNode<E>,
left:SearchTreeNode<E>,
right: SearchTreeNode<E>)
+addDupNode():

DupNode

+next:DupNode
+Index:Integer

+DupNode(Index:Integer)