Item
itemName: String combineKey: String itemDescript: String inLocationDescript: String Type: int montaryValue: double Uses: int emptyCounterpart: EmptyItem Length: int
Item() Item(name: String, shortName: string, description: String, examine, Message:String, whatKind: int) Item(name: String, shortName: string, description: String, examine, Message:String, whatKind: int, faceValue: double) Item(name: String, shortName: string, description: String, examine, Message:String, whatKind: int, useAmount: int, whenEmpty: EmptyItem) Item(name: String, shortName: string, description: String, examine, Message:String, whatKind: int, howLong: int) removeItemUses():void SwitchWithEmpty():void printItemDescription(nameToCheck: String): void determineItemsExistance(nameToCheck: String): Item determineSamePlace(itemToCheck: String): Object[] takeItemWithCost(): boolean takeItem(itemToCheck: String): void buyItem(itemToCheck: String): void TransferItem(itemToCheck: String): void transferAll(): void displayMap():void checkConditions(): boolean determineEatFood(itemToEat: String): void eatFood():void

CombinedItem	EmptyItem
capture: boolean part1: item part2: item	
CombinedItem() CombinedItem(part1: item, part2: item) combineItems (itemsToCombine: String): CombinedItem determineRemoveItems (item1: Item, item2: Item): void splitItem(itemToDump: String): void	EmptyItem() EmptyItem(name: String, shortName: string, description: String, examine, Message:String, whatKind: int)

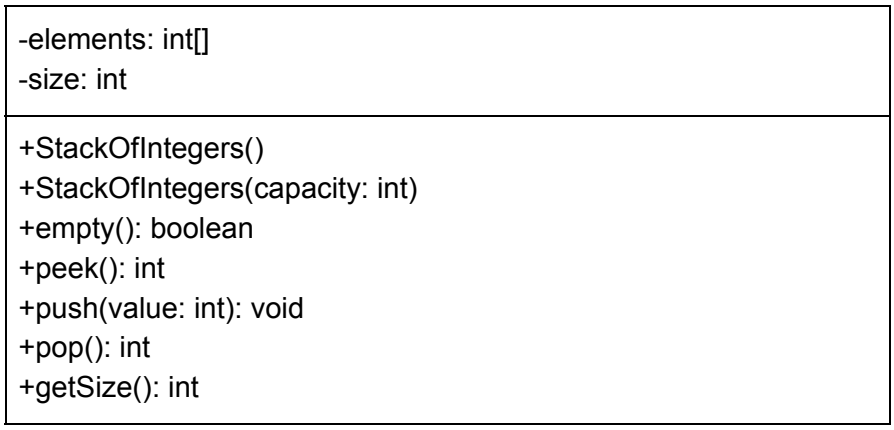
Person
locationIndex: int renderLocation: boolean name: String inventory: ArrayList<Item> MAX_INVENTORY: int inventoryEnabled: boolean moneyHad: double
Person () Person(intLocation: int) getName(): String moveLocation (direction: int): void renderLocation (): void nothingThere (): void goBack (): void driveThere(): void renderCurrentLocation(): void enableInventory (): void displayInventory(): void displayPersonalInventory():void displayCombinedItems(): void displayMoney(): String determineAddItem (itemToAdd: String): void addItem(itemToAdd: Item): void dropItem(itemToCheck: String):void stealItem(itemToSteal: String):void spendMoney (itemToBuy: Item): boolean AddCombinedItem(itemsToCombine: String): void checkWrench(maybeWrench: Item): boolean playerReset(): void

Container
contName: String contDescription: String capacity: int contents: ArrayList<Item>
Container() Container(name: String, description: String, canHold: int) determineContainerValid (nameToCheck: String): Container determineSamePlace(nameToCheck: String): boolean printContainerContents(nameToCheck: String): void addItemToContainer (itemToAdd: Item): void removeItem(thisItem: Item): void setItemsInContainers(): void emptyContainer(): void

SpecialtyContainer
Egg: boolean
SpecialtyContainer() SpecialtyContainer(name: String, description: String, canHold: int) ChangeEgg(): void printContainerContents(): void deployItem(itemToCheck: String): void getMultFactor(): int determineReturn(): Item GetFoundItem(): void alternateDeploy(itemDeployed:Item): boolean navigate(): void

Location
locationName: String locationDescription: String altDescription: String isVisited: boolean Items: ArrayList<Item> receptacle: ArrayList<Container> drivable: boolean
Location() Location(placeName: String, placeDescript: String, altDescript: String) Location(placeName: String, placeDescript: String, altDescript: String, canDrive:boolean) changeVisited(): void addContainer(toAdd: Container): void addItem(toAdd: Item): void examine():void getDrivableLocations(): void determineLocationExistence(locationName: String): int examineItems(): void examinePaidItems(): boolean isItEmpty(): boolean printItemDescriptions(): void examineContainers(): void findSpecialtyContainers(): boolean getMauled(): boolean setItemsAndContainers(): void wipeLocations(): void

StackOfIntegers


-elements: int[]
-size: int
+StackOfIntegers()
+StackOfIntegers(capacity: int)
+empty(): boolean
+peek(): int
+push(value: int): void
+pop(): int
+getSize(): int