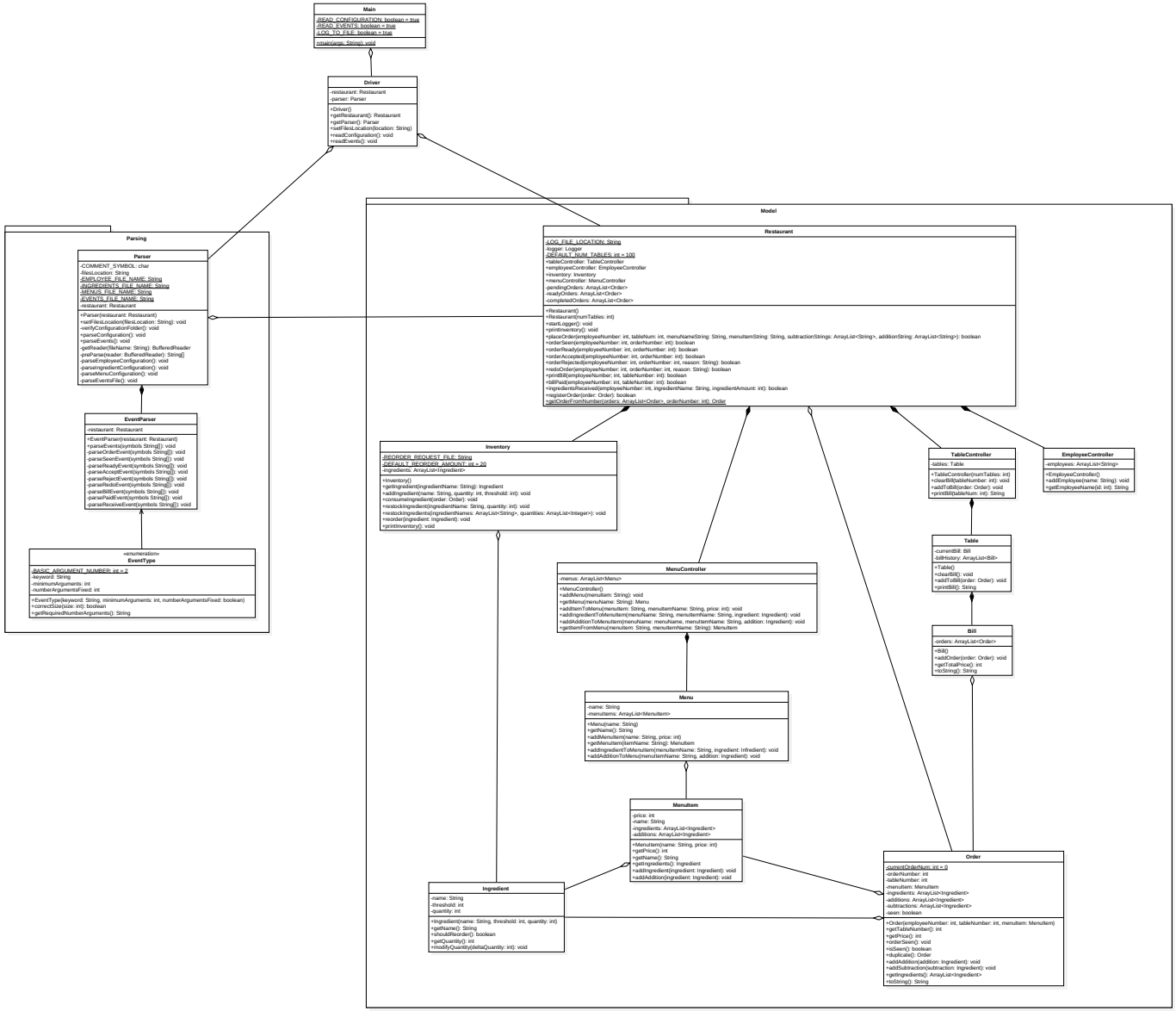
Main
-READ_CONFIGURATION: boolean = true
-READ_EVENTS: boolean = true
-LOG_TO_FILE: boolean = true
+main(args: String): void
Driver
-restaurant: Restaurant
-parser: Parser
+Driver()
+getRestaurant(): Restaurant
+getParser(): Parser
+setFilesLocation(location: String)
+readConfiguration(): void
+readEvents(): void
Parsing
Parser
-COMMENT_SYMBOL: char
-filesLocation: String
-EMPLOYEE_FILE_NAME: String
-INGREDIENTS_FILE_NAME: String
-MENUS_FILE_NAME: String
-EVENTS_FILE_NAME: String
-restaurant: Restaurant
+Parser(restaurant: Restaurant)
+setFilesLocation(filesLocation: String): void
-verifyConfigurationFolder(): void
+parseConfiguration(): void
+parseEvents(): void
-getReader(fileName: String): BufferedReader
-preParse(reader: BufferedReader): String[]
-parseEmployeeConfiguration(): void
-parseIngredientConfiguration(): void
-parseMenuConfiguration(): void
-parseEventsFile(): void
EventParser
-restaurant: Restaurant
+EventParser(restaurant: Restaurant)
+parseEvents(symbols String[]): void
-parseOrderEvent(symbols String[]): void
-parseSeenEvent(symbols String[]): void
-parseReadyEvent(symbols String[]): void
-parseAcceptEvent(symbols String[]): void
-parseRejectEvent(symbols String[]): void
-parseRedoEvent(symbols String[]): void
-parseBillEvent(symbols String[]): void
-parsePaidEvent(symbols String[]): void
-parseReceiveEvent(symbols String[]): void
«enumeration»
EventType
-BASIC_ARGUMENT_NUMBER: int = 2
-keyword: String
-minimumArguments: int
-numberArgumentsFixed: int
+EventType(keyword: String, minimumArguments: int, numberArgumentsFixed: boolean)
+correctSize(size: int): boolean
+getRequiredNumberArguments(): String
Model
Restaurant
-LOG_FILE_LOCATION: String
-logger: Logger
-DEFAULT_NUM_TABLES: int = 100
+tableController: TableController
+employeeController: EmployeeController
+inventory: Inventory
+menuController: MenuController
-pendingOrders: ArrayList<Order>
-readyOrders: ArrayList<Order>
-completedOrders: ArrayList<Order>
+Restaurant()
+Restaurant(numTables: int)
+startLogger(): void
+printInventory(): void
+placeOrder(employeeNumber: int, tableNum: int, menuNameString: String, menuItemString: String, subtractionStrings: ArrayList<String>, additionString: ArrayList<String>): boolean
+orderSeen(employeeNumber: int, orderNumber: int): boolean
+orderReady(employeeNumber: int, orderNumber: int): boolean
+orderAccepted(employeeNumber: int, orderNumber: int): boolean
+orderRejected(employeeNumber: int, orderNumber: int, reason: String): boolean
+redoOrder(employeeNumber: int, orderNumber: int, reason: String): boolean
+printBill(employeeNumber: int, tableNumber: int): boolean
+billPaid(employeeNumber: int, tableNumber: int): boolean
+ingredientsReceived(employeeNumber: int, ingredientName: String, ingredientAmount: int): boolean
+registerOrder(order: Order): boolean
+getOrderFromNumber(orders: ArrayList<Order>, orderNumber: int): Order
Inventory
-REORDER_REQUEST_FILE: String
-DEFAULT_REORDER_AMOUNT: int = 20
-ingredients: ArrayList<Ingredient>
+Inventory()
+getIngredient(ingredientName: String): Ingredient
+addIngredient(name: String, quantity: int, threshold: int): void
+consumeIngredient(order: Order): void
+restockIngredient(ingredientName: String, quantity: int): void
+restockIngredient(ingredientNames: ArrayList<String>, quantities: ArrayList<Integer>): void
+reorder(ingredient: Ingredient): void
+printInventory(): void
TableController
-tables: Table
+TableController(numTables: int)
+clearBill(tableNumber: int): void
+addToBill(order: Order): void
+printBill(tableNum: int): String
EmployeeController
-employees: ArrayList<String>
+EmployeeController()
+addEmployee(name: String): void
+getEmployeeName(id: int): String
Table
-currentBill: Bill
-billHistory: ArrayList<Bill>
+Table()
+clearBill(): void
+addToBill(order: Order): void
+printBill(): String
MenuController
-menus: ArrayList<Menu>
+MenuController()
+addMenu(menuItem: String): void
+getMenu(menuName: String): Menu
+addItemToMenu(menuItem: String, menuItemName String, price: int): void
+addIngredientToMenuItem(menuName String, menuItemName String, ingredient: Ingredient): void
+addAdditionToMenuItem(menuName: menuName, menuItemName String, addition: Ingredient): void
+getItemFromMenu(menuItem: String, menuItemName String): MenuItem
Bill
-orders: ArrayList<Order>
+Bill()
+addOrder(order: Order): void
+getTotalPrice(): int
+toString(): String
Menu
-name: String
-menuItems: ArrayList<MenuItem>
+Menu(name: String)
+getName(): String
+addMenuItem(name: String, price: int)
+getMenuItem(itemName: String): MenuItem
+addIngredientToMenuItem(menuItemName: String, ingredient: Infredient): void
+addAdditionToMenu(menuItemName: String, addition: Ingredient): void
MenuItem
-price: int
-name: String
-ingredients: ArrayList<Ingredient>
-additions: ArrayList<Ingredient>
+MenuItem(name: String, price: int)
+getPrice(): int
+getName(): String
+getIngredients(): Ingredient
+addIngredient(ingredient: Ingredient): void
+addAddition(ingredient: Ingredient): void
Ingredient
-name: String
-threshold: int
-quantity: int
+Ingredient(name: String, threshold: int, quantity: int)
+getName(): String
+shouldReorder(): boolean
+getQuantity(): int
+modifyQuantity(deltaQuantity: int): void
Order
-currentOrderNum: int = 0
-orderNumber: int
-tableNumber: int
-menuItem: MenuItem
-ingredients: ArrayList<Ingredient>
-additions: ArrayList<Ingredient>
-subtractions: ArrayList<IngredientX>
-seen: boolean
+Order(employeeNumber: int, tableNumber: int, menuItem: MenuItem)
+getTableNumber(): int
+getPrice(): int
+orderSeen(): void
+isSeen(): boolean
+duplicate(): Order
+addAddition(addition: Ingredient): void
+addSubtraction(subtraction: Ingredient): void
+getIngredients(): ArrayList<Ingredient>
+toString(): String