Sample Project Question

1. Food Court Management Application:
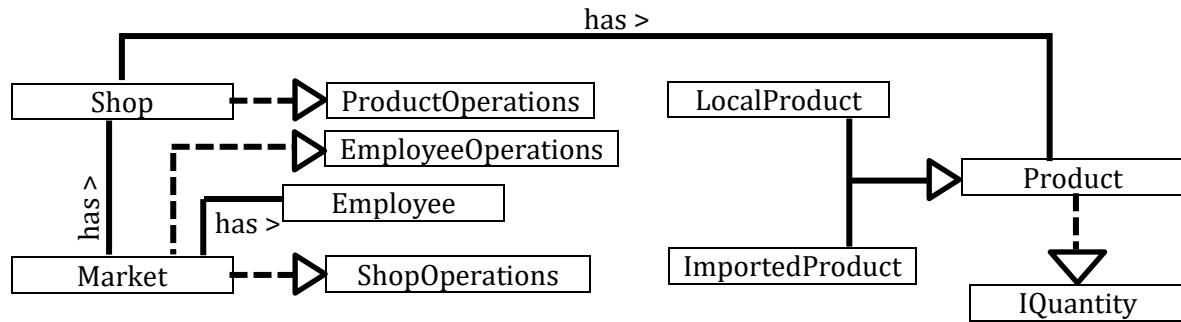

IQuantity

boolean addQuantity(int amount)
boolean sellQuantity(int amount)

FoodItemOperations
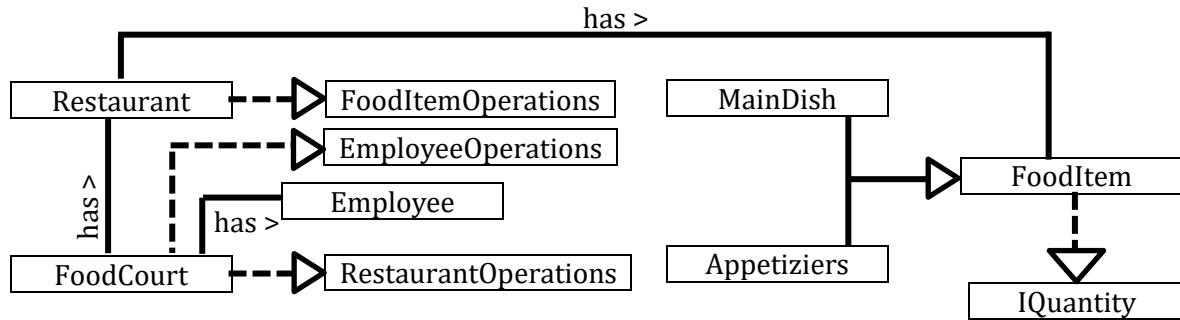
boolean insertFoodItem(FoodItem f)
boolean removeFoodItem(FoodItem f)
FoodItem searchFoodItem(String fid)
void showAllFoodItems( )

RestaurantOperations

boolean insertRestaurant(Restaurant r)
boolean removeRestaurant(Restaurant r)
Restaurant searchRestaurant(String rid)
void showAllRestaurants( )

FoodItem

String fid
String name
int availableQuantity
double price

void setFid(String fid)
void setName(String name)
void setAvailableQuantity(int quantity)
void setPrice(double price)
String getFid( )
String getName( )
int getAvailableQuantity( )
double getPrice( )
abstract void showInfo( )

Employee

String name
String empId
double salary

void setName(String name)
void setEmpId(String empId)
void setSalary(double salary)
String getName( )
String getEmpId( )
double getSalary( )

EmployeeOperations

boolean insertEmployee(Employee e)
boolean removeEmployee(Employee e)
Employee searchEmployee(String empId)
void showAllEmployees( )

MainDish

String category

void setCategory(String category)
String getCategory( )

Restaurant

String rid
String name
FoodItem fooditems[ ]

void setName(String name)
void setRid(String rid)
String getName( )
String getRid( )

Appitizers

String size

void setSize(String size)
String getSize( )

FoodCourt

Restaurant restaurants[ ]
Employee employees[ ]

Start

The **Start** class contains the main method. Inside the main method the application prompts the user to choose among the following options:

1. Employee Management
2. Restaurant Management
3. Restaurant FoodItem Management
4. FoodItem Quantity Add-Sell
5. Exit

Upon choosing any of the option the user is given some further options to choose from:

Options for Employee Management:

1. Insert New Employee   2. Remove Existing Employee   3. Show All Employees   4. Search an Employee   5. Go Back

Options for Restaurant Management:

1. Insert New Restaurant   2. Remove Existing Restaurant   3. Show All Restaurants   4. Search a Restaurant   5. Go Back

Options for Restaurant FoodItem Management:

1. Insert New FoodItem   2. Remove Existing FooItem   3. Show All FoodItems   4. Search a FoodItem   5. Go Back

Options for FoodItem Quantity Add-Sell:

1. Add FoodItem   2. Sell FoodItem   3. Show Add Sell History   4. Go Back

The application will prompt the user to give input for the necessary values as per the choice. The application will repeatedly go on according to the user choices. Whenever FoodItem is being added or sold, the information is written in a text file and when the user requests for the add-sell history, the information written in the text file is read and displayed on screen.

Sample Project Question

2. Market Management Application:

IQuantity

boolean addQuantity(int amount)
boolean sellQuantity(int amount)

ProductOperations

boolean insertProduct(Product p)
boolean removeProduct(Product p)
Product searchProduct(String pid)
void showAllProducts( )

ShopOperations

boolean insertShop(Shop s)
boolean removeShop(Shop s)
Shop searchShop(String sid)
void showAllShops( )

Product

String pid
String name
int availableQuantity
double price

void setPid(String pid)
void setName(String name)
void setAvailableQuantity(int quantity)
void setPrice(double price)
String getPid( )
String getName( )
int getAvailableQuantity( )
double getPrice( )
abstract void showInfo( )

Employee

String name
String empId
double salary

void setName(String name)
void setEmpId(String empId)
void setSalary(double salary)
String getName( )
String getEmpId( )
double getSalary( )

EmployeeOperations

boolean insertEmployee(Employee e)
boolean removeEmployee(Employee e)
Employee searchEmployee(String empId)
void showAllEmployees( )

LocalProduct

double discountRate

void setDiscountRate(double discountRate)
double getDiscountRate( )

Shop

String sid
String name
Product products[ ]

void setName(String name)
void setSid(String sid)
String getName( )
int getNid( )

ImportedProduct

String countryName

void setCountryName(String countryName)
String getCountryName( )

Market

Shop shops[ ]
Employee employees[ ]

Start

The **Start** class contains the main method. Inside the main method the application prompts the user to choose among the following options:

1. Employee Management
2. Shop Management
3. Shop Product Management
4. Product Quantity Add-Sell
5. Exit

Upon choosing any of the option the user is given some further options to choose from:

Options for Employee Management:

1. Insert New Employee   2. Remove Existing Employee   3. Show All Employees   4. Search an Employee   5. Go Back

Options for Shop Management:

1. Insert New Shop   2. Remove Existing Shop   3. Show All Shops   4. Search a Shop   5. Go Back

Options for Shop Product Management:

1. Insert New Product   2. Remove Existing Product   3. Show All Product   4. Search a Product   5. Go Back

Options for Product Quantity Add-Sell:

1. Add Product   2. Sell Product   3. Show Add Sell History   4. Go Back

The application will prompt the user to give input for the necessary values as per the choice. The application will repeatedly go on according to the user choices. Whenever Product is being added or sold, the information is written in a text file and when the user requests for the add-sell history, the information written in the text file is read and displayed on screen.

Sample Project Question

Requirements:

- Object Oriented Programming Principles must be followed.
- The whole program must be organized using at least three user defined packages.
- File Read Write must be there as mentioned above.
- The program must be able to handle exceptions by itself.

Project Submission Instruction:

- It is a group project and every group must contain exactly 2 members. No more. No less.
- Fill up the form in the following link:
  Link: https://qrgo.page.link/Qr7eJ
- I will assign a GroupID to your group, allocate a slot for your project defense and viva, and upload it in VUES.
- You will have to Zip and Upload the project in your VUES account before your allocated schedule. The name of your zipped folder will be according to your GroupID.