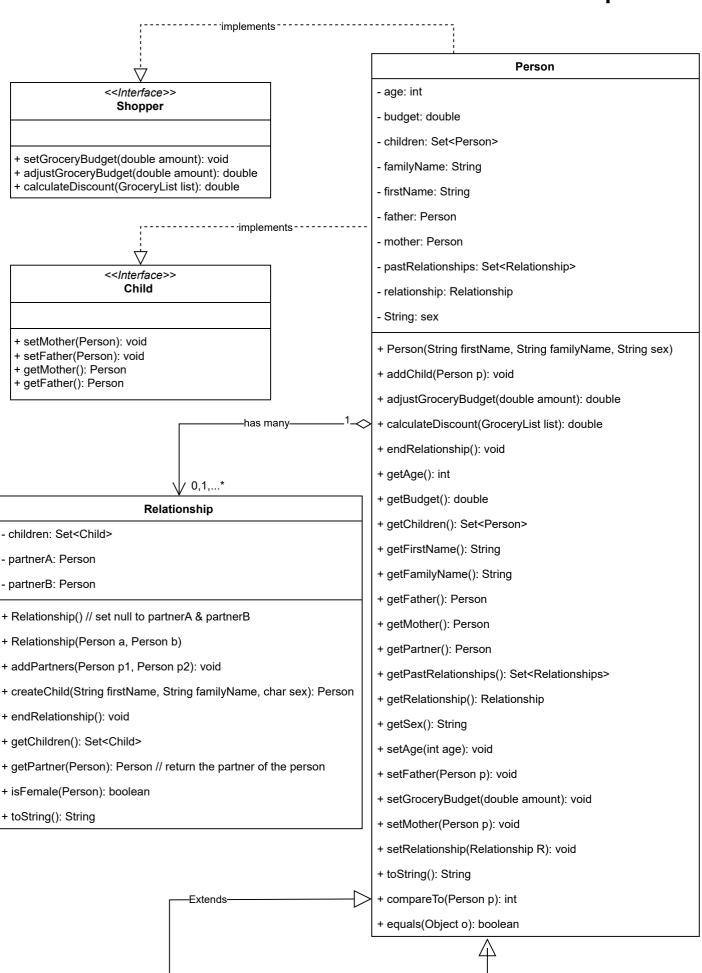
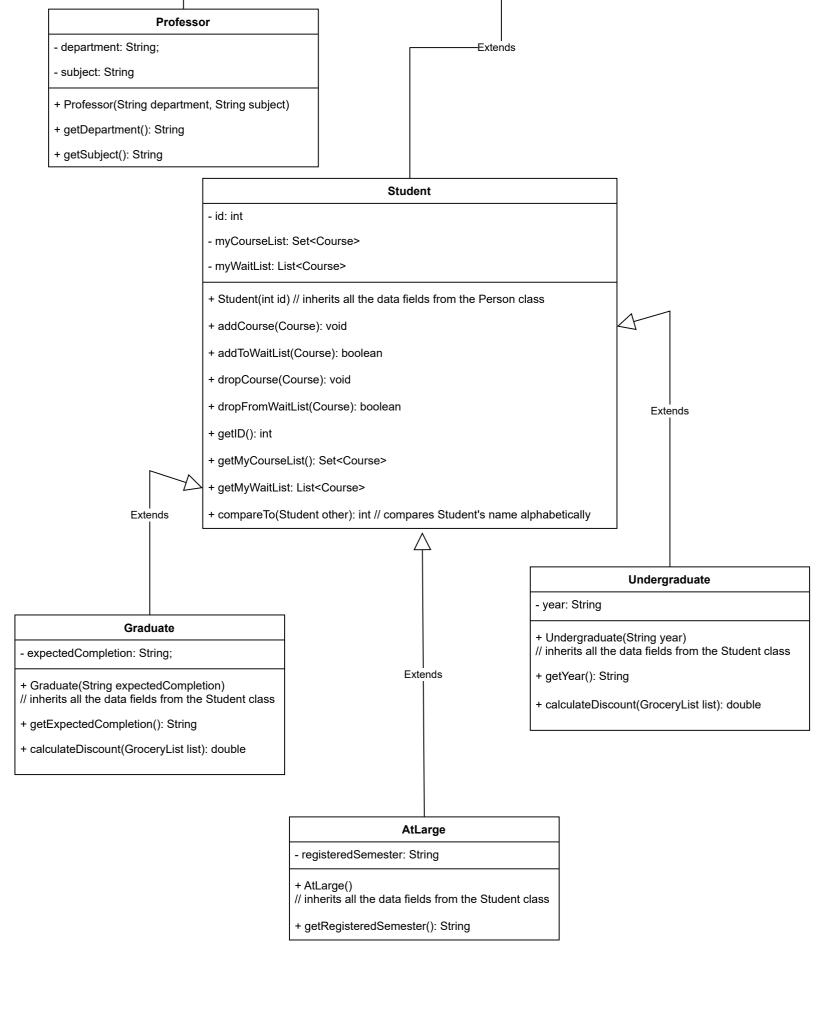
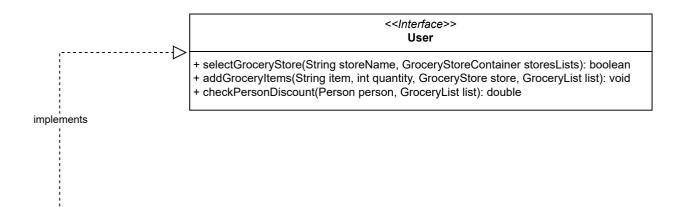


Person-Professor-Students-Relationship





Registry



Registry

- courseFinder: Map<String, Course>

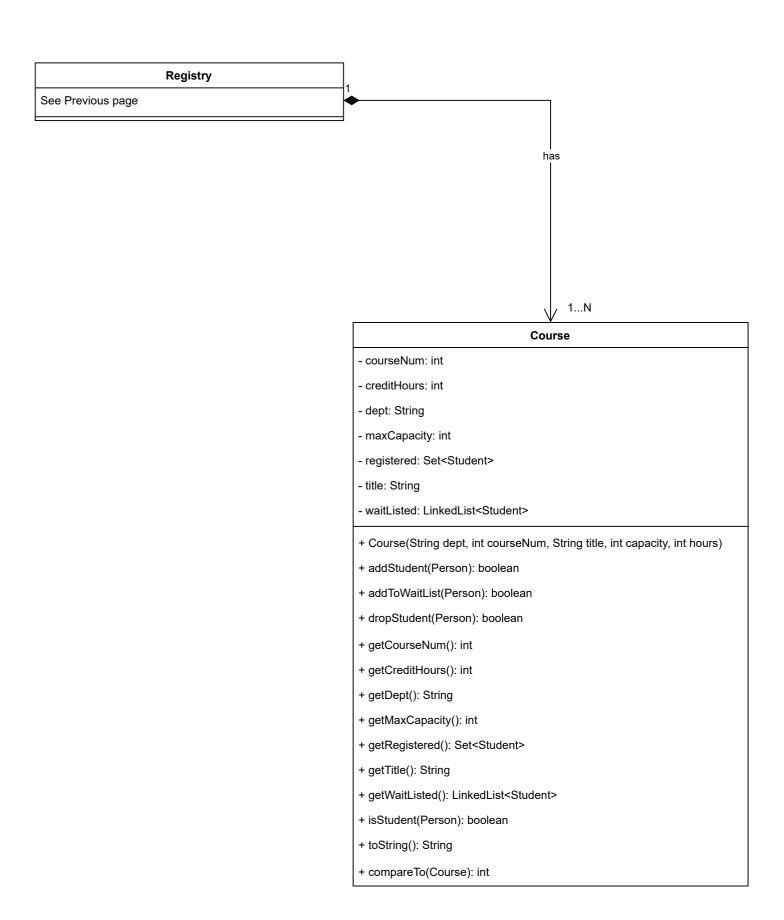
- courses: ArrayList<Course>

- registeredPeople: Map<String, Person>

+ Registry(): //constructor

- + addGroceryItems(String itemName, int quantity, GroceryStore store, GroceryList list): void
- + addPerson(Person): void
- + checkPersonDiscount(Person, GroceryList): double
- + enrollStudent(String firstName, String familyName, String dept, int cNum): void
- + getAllChildren(Person): ArrayList<Person>
- + getCourse(String dept, int cNum): Course
- + getCourses(): ArrayList<Course>
- + getMaternalLine(Person): ArrayList<Person>
- + getPaternalLine(Person): ArrayList<Person>
- + getPerson(String firstName, String familyName, String sex): Person
- + makeNewPerson(String firstName, String familyName, String sex): Person
- + recordCourse(String dept, int cNum, String title, int maxCap, int hours): void
- + removeStudent(String firstName, String familyName, String dept, int cNum): void
- + selectGroceryStore(String storeName, GroceryStoreContainer storeLists): boolean
- + toString(): String

Course



Grocery **GroceryStoreContainer** - groceryStores: Map<String, GroceryStore> + GroceryStoreContainer(): //constructor + addGroceryStore(GroceryStore store): void + containsGroceryStore(String storeName): boolean + getGroceryStore(String storeName): GroceryStore + removeGroceryStore(GroceryStore store): void contains 1...N GroceryList **GroceryStore** - groceryList: ArrayList<GroceryItemOrder> - storeName: String - totalCost: double - groceryItems: List<GroceryItem> + GroceryStore(String storeName) + GroceryList(): //constructor + addItem(GroceryItemOrder item): void ¹ + addItem(GroceryItem item): void has a + getGroceryList(): ArrayList<GroceryItemOrder> + containsItem(GroceryItem item): boolean + getListLength(): int + getItemCost(): List<GroceryItem> + getTotalCost(): double + getItemName(): List<GroceryItem> + removeItem(GroceryItemOrder item): void + getStoreName(): String + toString(): String + removeItem(GroceryItem item): void + toString(): String has items 1...N GroceryItemOrder GroceryItem - quantity: int - item: String - pricePerUnit: double + GroceryItemOrder(int quantity) Extends // inherits the data fields from GroceryItem + GroceryItem(String item, double price): //constructor + getQuantity(): int + getItem(): String + getPricePerQuantity(): double + getPricePerUnit(): double + setItem(String item): void + toString(): String

Final UML

<<Interface>> User

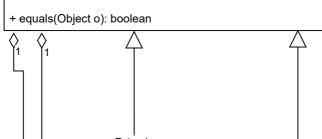
- + selectGroceryStore(String storeName, GroceryStoreContainer storesLists): boolean
- + addGroceryItems(String item, int quantity, GroceryStore store, GroceryList list): void
- + checkPersonDiscount(Person person, GroceryList list): double
- 1...N searches implements

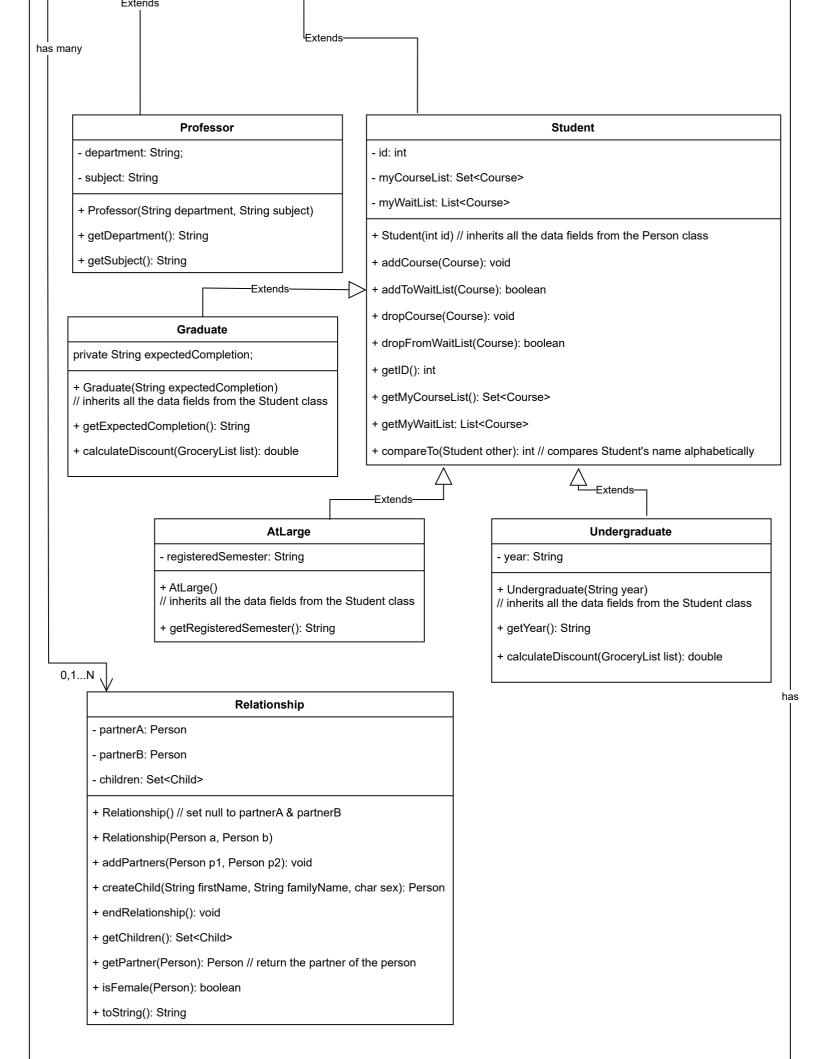
Registry

- courseFinder: Map<String, Course>
- courses: ArrayList<Course>
- registeredPeople: Map<String, Person>
- + Registry(): //constructor
- + addGroceryItems(String itemName, int quantity, GroceryStore store, GroceryList list): void
- + addPerson(Person): void
- + addPerson(Person): void
- + checkPersonDiscount(Person, GroceryList): double
- + enrollStudent(String firstName, String familyName, String dept, int cNum): void
- + getAllChildren(Person): ArrayList<Person>
- + getCourse(String dept, int cNum): Course
- + getCourses(): ArrayList<Course>
- + getMaternalLine(Person): ArrayList<Person>
- + getPaternalLine(Person): ArrayList<Person>
- $+ \ getPerson(String \ firstName, \ String \ familyName, \ String \ sex): \ Person$
- + makeNewPerson(String firstName, String familyName, String sex): Person
- + recordCourse(String dept, int cNum, String title, int maxCap, int hours): void
- + removeStudent(String firstName, String familyName, String dept, int cNum): void
- $+\ select Grocery Store (String\ store Name,\ Grocery Store Container\ store Lists):\ boolean$
- + toString(): String

Person

- age: int
- budget: double
- children: Set<Person>
- familyName: String
- firstName: String
- father: Person
- mother: Person
- pastRelationships: Set<Relationship>
- relationship: Relationship
- String: sex
- + Person(String firstName, String familyName, String sex)
- + addChild(Person p): void
- + adjustGroceryBudget(double amount): double
- + calculateDiscount(GroceryList list): double
- + endRelationship(): void
- + getAge(): int
- + getBudget(): double
- + getChildren(): Set<Person>
- + getFirstName(): String
- + getFamilyName(): String
- + getFather(): Person
- + getMother(): Person
- + getPartner(): Person
- + getPastRelationships(): Set<Relationships>
- + getRelationship(): Relationship
- + getSex(): char
- + setAge(int age): void
- + setFather(Person p): void
- + setGroceryBudget(double amount): void
- + setMother(Person p): void
- + setRelationship(Relationship R): void
- + toString(): String
- + compareTo(Person p): int







- courseNum: int

- creditHours: int

- dept: String

- maxCapacity: int

- registered: Set<Student>

- title: String

- waitListed: LinkedList<Student>

+ Course(String dept, int courseNum, String title, int capacity, int hours)

Course

+ addStudent(Person): boolean

+ addToWaitList(Person): boolean

+ dropStudent(Person): boolean

+ getCourseNum(): int

+ getCreditHours(): int

+ getDept(): String

+ getMaxCapacity(): int

+ getRegistered(): Set<Student>

+ getTitle(): String

+ getWaitListed(): LinkedList<Student>

+ isStudent(Person): boolean

+ toString(): String

+ compareTo(Course): int

GroceryList

- groceryList: ArrayList<GroceryItemOrder>

- totalCost: double

+ GroceryList(): //constructor

+ addItem(GroceryItemOrder item): void

+ getGroceryList(): ArrayList<GroceryItemOrder>

+ getListLength(): int

+ getTotalCost(): double

+ removeItem(GroceryItemOrder item): void

+ toString(): String

GroceryStoreContainer

- groceryStores: Map<String, GroceryStore>

+ GroceryStoreContainer(): //constructor

+ addGroceryStore(GroceryStore store): void

+ containsGroceryStore(String storeName): boolean

+ getGroceryStore(String storeName): GroceryStore

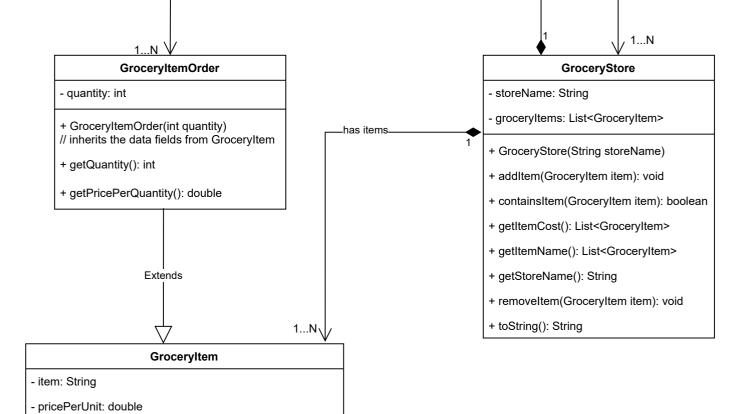
contains

+ removeGroceryStore(GroceryStore store): void

+ removeGrocerySto

has I

1...N



+ GroceryItem(String item, double price): //constructor

+ getItem(): String

+ toString(): String

+ getPricePerUnit(): double+ setItem(String item): void