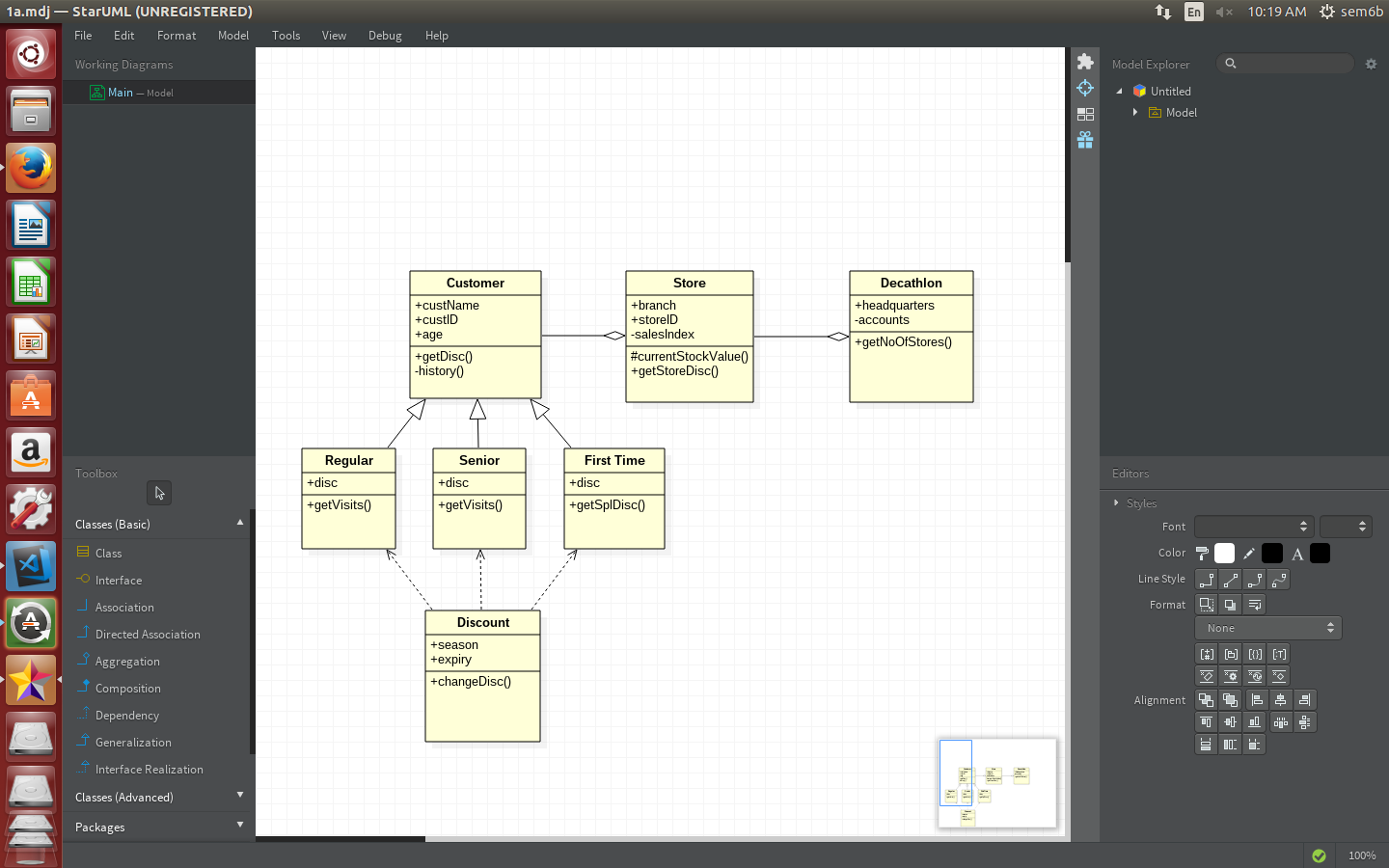
Ramaiah Institute of Technology

Department of Information Science and Engg.

**PART-A**

***Case-study to understand the limitation of traditional Object Oriented Design and appreciate need for DesignPatterns. Use UML Notations to design.***

You are a fresh analyst deputed to design the software for Decathlon Chain of Stores in Karnataka. You are informed about the Business Logic of Point of Sales criteria by Ms.Veronica Lodge, a dynamic business tycoon operating out of Decathlon Mumbai. She informs you that there are different types of Customers of Decathlon namely, Regular Customers, Senior Citizens and First Time Customers. Regular Customers are given a discount of 12%, Senior Citizens 10% and First Time Customers 15%. Apart from this, based on the sales-index of previous day, a Store-level discount is determined every day. This is dynamic. **E.g**.Rs.100 off for every purchase above Rs.2000. Using the Object Oriented Principles of Encapsulation, Abstraction, Inheritance, Composition and Aggregation that you have studied until this semester, give at least two ways to design this system.

****

**Common Case Study for Q#1 to Q#9**

‘Decathlon’ is a Sports retail-store started in France. Today it spreads across 22 countries &amp; has900 outlets in these countries. It has a ‘Point of Sale’ software system called ‘Decathlon POS’, which uses various kinds of 3rd-party software sourced locally from the various countries they are established. You are a software consultant for Decathlon, in Bangalore, with a team of consultants reporting to you. When you analyze your answer for choosing a pattern, explain

wherever applicable, keeping in mind the following four design principles:

 Separation of concerns

 Program to an interface, not a concrete implementation

 Prefer composition over inheritance

 Open-Close principle (Open for extension, Closed for modification)

1. Adaptor (Structural): To establish the 1st Decathlon store in Mauritius, you go along

with Mr. Satya Nadella,an expert in finding 3rd-party partners. For e.g. a 3rd-party Tax-

Calculator system to cater to the specifics of Salesand VAT (Value-added services Tax)

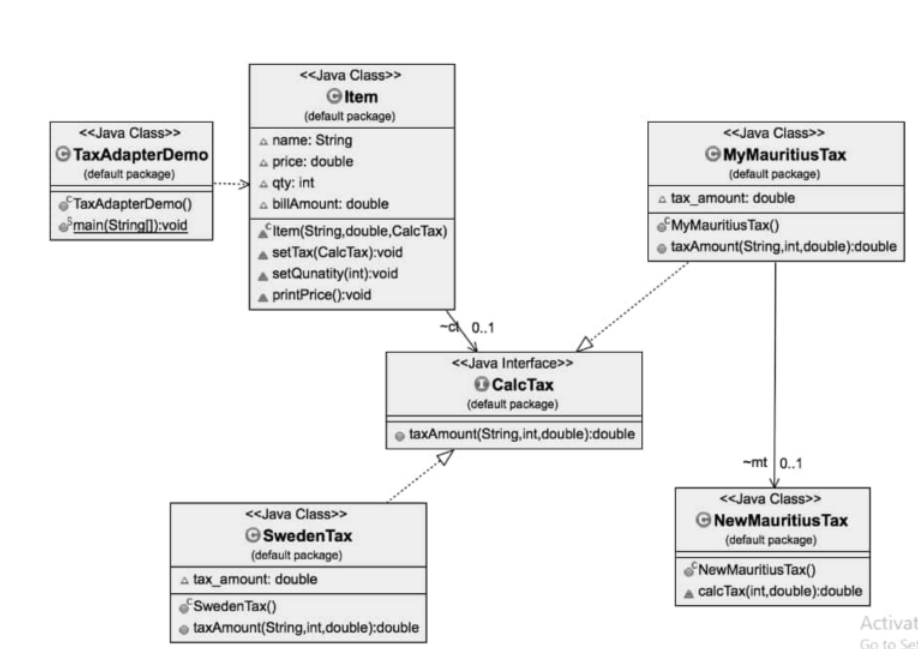
tax calculations in different countries. He finds a 3rd-party Tax-Calculatorsystem called

‘MauriTax’ in Port Louis. The problem is, the APIs used by ‘MauriTax’ for tax-

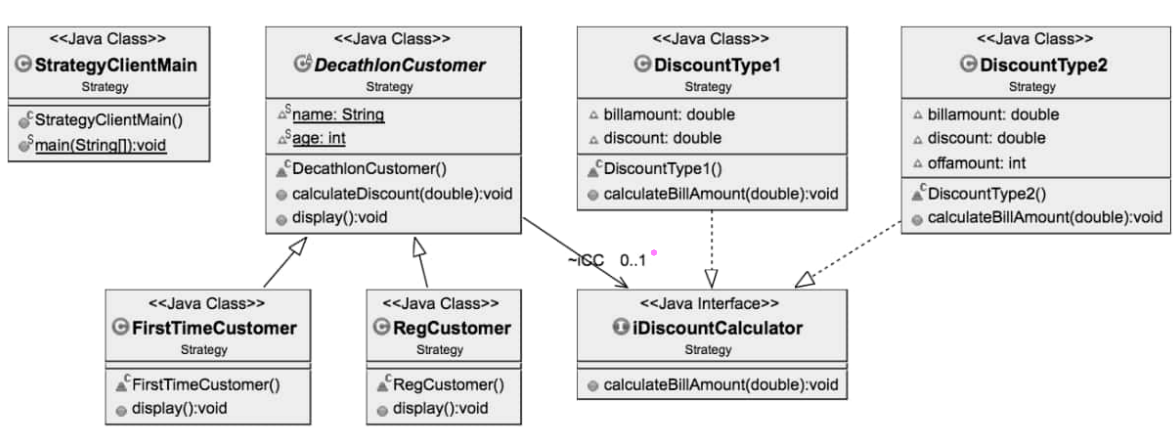
calculation is fixed&amp; cannot be changed. The ‘MauriTax’ APIs are incompatible with

‘Decathlon POS’.

How will you use the Adaptor Pattern to design &amp; implement?

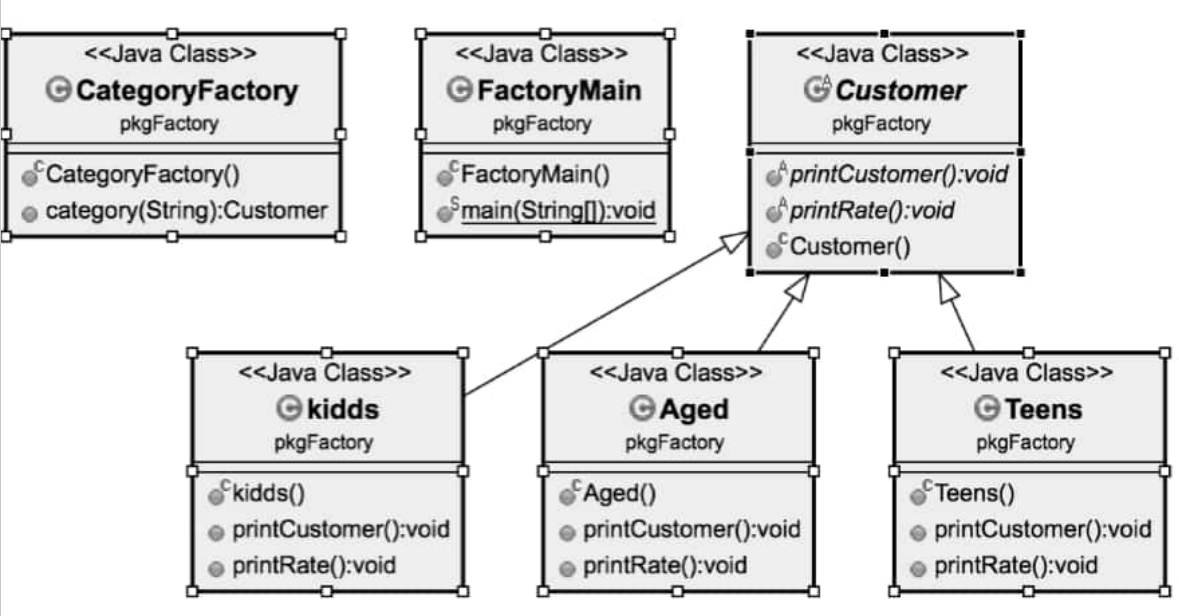


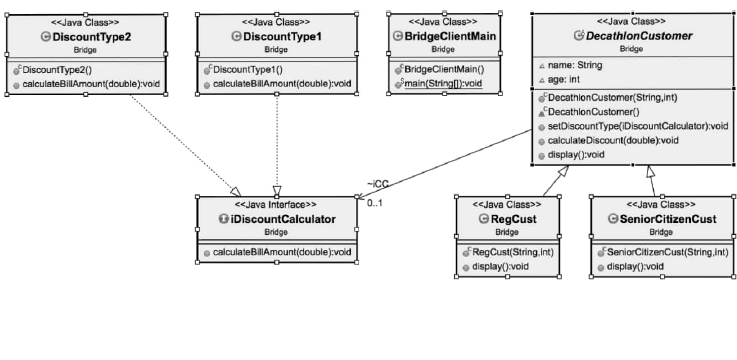
**Strategy (Behavioural):** How will you use the Strategy Pattern to tackle the limitations of traditional ObjectOriented Design highlighted in PART A? *The design must handle varying price-schemes having different pricingalgorithms.* Design & implement



1. **Factory Method (Creational):** The ‘Decathlon POS’ software system classifies its customers as senior-citizens(60 and above), First-Time customers, Regular Customers. There is a very high possibility that the CustomerType hierarchy will vary, depending upon the sales-pattern. **For e.g.** there could be the need to introduce newcategories based on the customer gender, different age groups for kids (0-5, 6-12), teenagers (13-19) and agegroups between 20 to 60(Twenties, 30s, 40s and 50s).

You are advised by Mr.Sundar Pichai, the technical architect of your team, whom you trust, to use Factory MethodPattern in order to instantiate the above Customer Type hierarchy of concrete implementation of objects. Designand implement using this.



4.Bridge (Structural): You get a call from Ms.Masaba Gupta of Bangalore Decathlon office that there is a policy decision made globally to introduce discount slabs for a whole month twice in a year. The discount month will be in January and July after reviewing the sales made from Feb to June (first five months) and Aug to December (last five months) respectively. It is decided to provide four slabs of discounts in 2017, namely, 30%, 25%, 20% and 15%, based on the sports item purchased. For e.g. all tennis rackets could have a 20% discount while cricket bats could only have a 15% discount. All exercise tread-mills could be given a 30% discount while boxing-gloves could have a 25% discount. Point to be noted here is that, the slabs of discount may not remain the same in 2018. It is likely to vary year after year. The ‘Decathlon POS’ software system classifies its customers as Senior-Citizens (60 and above), First-Time Customers, Regular Customers as of now. There is a very high possibility that the Customer Type hierarchy will vary, depending upon the sales-pattern. For e.g. there could be the need to introduce new categories based on the customer gender. Use the Bridge Pattern to design & implement, so that both the Customer Type hierarchy of classes as well as the Discount Percentage hierarchy of classes can both vary independently? That is, they are not tied to each other

5. Observer (Behavioural): There will be different discounts being offered for the sports items in Decathlon Stores across the globe for different festivals being celebrated in the various countries these stores are established. Assume that the Decathlon Chain of Stores fixes a particular discount slab for its items for a festival of a country. Use the Observer Pattern to design and implement a system to notify the customers of the Decathlon stores of that country about the various festival / seasonal discount rates as and when they are announced.



6. Façade (Structural): You get a call from Ms.Betty Cooper of Bangalore Decathlon office that there is a policy decision made globally to incorporate some new rules for ‘Process Sale UseCase’. For e.g. if payment is made via gift-certificate, the customer can buy only one item for the amount in the certificate. No other items can be bought with that gift-certificate. There must be no cash-back to the customer if the item costs less than amount specified in the giftcertificate. If the item costs more, the excess payment can be accepted via cash only & not credit / debit cards. When a new sale is created, these rules must become effective. You come to know from Mr. Satya Nadella, an expert in finding 3rd-party partners, that the

Italian Competitor for Decathlon called ‘Sport 2000’ has a ready-made ‘rule-engine’ sub-system for this, whose specific implementation details is not known yet, as the business heads of Decathlon & Sport 2000 are chalking out the software purchase terms. This Sport 2000 rule-engine will be responsible for evaluating a set of rules against an operation & indicating if any of the rules invalidated the operation (e.g. ‘makeNewSale’ operation).

How will you use the Façade pattern to provide a common unified interface to a dissimilar set of implementations, developed by a 3rd-party vendor, the implementation details are not known to you?

