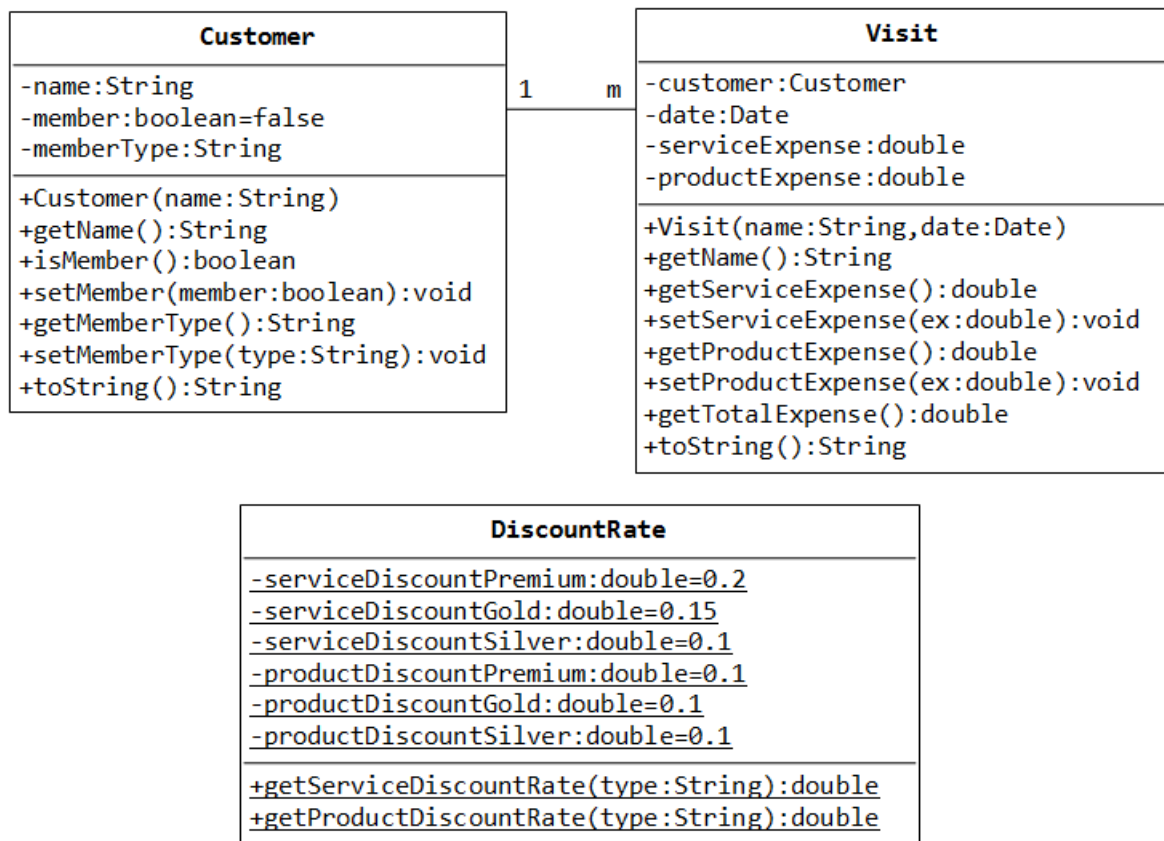


Ćwiczenie 8

Napiszmy system dla salonu piękności, który zapewnia usługi i sprzedaje produkty kosmetyczne. Oferowane są trzy tryby członkostwa: Premium, Gold i Silver. Każdy z nich odpowiednio dostaje rabat w postaci 20%, 15% oraz 10% dla członkostwa Silver. Klienci bez członkostwa nie posiadają zniżek. Wszyscy klienci otrzymują stałą zniżkę 10% na zakup produktów (procent zniżki może ulec zmianie w przyszłości). System powinien posiadać trzy klasy: *Customer*, *DiscountRate* i *Visit*, które pokazano na diagramie. Powinien także obliczyć łączny rachunek, jeżeli klient zakupi \$x\$ produktów i \$y\$ usług w przeciągu jednej wizyty.



Dla podanego diagramu, utwórz w salonie kilku klientów oraz przetestuj działanie na wywoływaniu różnych cen produktów i usług.

Ćwiczenie 9

Utwórzmy diagram klas oraz stwórzmy wybrane klasy do obsługi systemu kolejkowego na poczcie lub w banku. Klienci powinni mieć możliwość pobierania kolejnych numerków. Numerki wydawane są w zależności od oddziałów, które mają przyporządkowane literki. Przykładowo:

A - Oddział #1

B - Oddział #2

Numerki mogą być wydawane w sposób następujący:

A1, A2, B1, B2, B3, B4, A3, B5 itd. w zależności od wybranego oddziału

Stanowiska pracownicze (okienka) mogą obsługiwać różne oddziały np. A, B, C, inne stanowisko może obsługiwać tylko A i D.

Wszystkie stanowiska powinny dziedziczyć po wspólnej klasie stanowisk.

Celem aplikacji, jest stworzenie instancji klasy, która pozwoli na wywoływanie kolejnych klientów np. za pomocą metody `.nextCustomer()`, która będzie zwracała "okienko", do którego może podejść klient. Zwróć uwagę na to, że okienka powinny mieć dostęp do 'listy' klientów oczekujących.