

ЗАДАЧИ

1. Абстрактный класс *Account* имеет два дочерних класса: счет физического лица *PersonalAccount* и юридического *CompanyAccount*. При решении задачи используйте диаграммы классов.

а. Добавьте класс *Person* с общедоступным атрибутом *FullName* строкового типа и свяжите его с классом *PersonalAccount* ассоциацией *Owns* с плюсом *owner* у *Person* и навигируемым плюсом *account* у *PersonalAccount*.

б. Аналогично для счета юридического лица добавьте владельца *Company*, свяжите анонимной ассоциацией с *CompanyAccount* и укажите подходящие названия полюсов.

в. Добавьте класс адреса *Address* с атрибутами строкового типа *street*, *city* и целочисленным положительным *building*. Укажите с помощью новых анонимных ассоциаций, что *Person* может иметь адрес регистрации *registeredAt*, фактический адрес *actual*, в то время как компания связана с одним юридическим адресом *legalAddress* и может иметь почтовый адрес *postAddress*.

2. Интерфейс *Stack* определяет операции помещения в стек *push* с параметром *obj* типа *Element*, операцию получения элемента из стека *pop* с возвращаемым значением типа *Element*. При решении задачи используйте диаграммы классов.

а. Добавьте в интерфейс *Stack* операции очистки стека *reset*, которая не имеет параметров, статическую операцию создания нового стека *createNew* с возвращаемым значением типа *Stack*.

б. Покажите, что интерфейс *Stack* зависит от типа данных *Element*.

в. Добавьте класс *ListStack*, который реализует интерфейс *Stack*. Покажите реализуемые классом операции интерфейса.

г. Добавьте в класс *ListStack* частное структурное свойство *arr* типа *Element* с кратностью больше нуля, значения которого упорядочены и могут повторяться.

д. Добавьте частный целочисленный атрибут *increment* только для чтения и защищенную операцию изменения размера *resize* с целочисленным параметром *newSize*.

е. Покажите на диаграмме экземпляров экземпляр *stack* класса *ListStack*, свойство *arr* которого содержит элемент *first* типа *Element* первым и *second* того же типа вторым. Укажите, что атрибут *increment* экземпляра *stack* равен 10.

3. В пространстве имен *Time* расположены перечисления *Month*, *DayOfWeek*, а также классы *Date* и *Period*. При решении задачи используйте диаграммы классов.

а. Укажите, что перечисление *Month* может принимать значения: *Jan*, *Feb*, *Mar*, *Apr*, *May*, *Jun*, *Jul*, *Aug*, *Sep*, *Oct*, *Nov*, *Dec*.

б. Укажите, что перечисление *DayOfWeek* может принимать значения: *Mon*, *Tue*, *Wed*, *Thu*, *Fri*, *Sat*, *Sun*.

в. Добавьте классу *Date* частные атрибуты *year*, *month*, *dayOfMonth* типа *Integer*, а также общедоступные операции:

– получения года *getYear* типа *Integer*; – получения месяца *getMonth* типа *Month*; – получения дня *getDayOfMonth* типа *Integer*; – получения дня недели *getDayOfWeek* типа *DayOfWeek*.

г. Добавьте классу *Date* общедоступную статическую операцию *now()* типа *Date*.

д. Добавьте классу *Period* общедоступную статическую операцию *between*. У операции два аргумента: *from* и *to*. Оба аргумента имеют тип *Date*. Операция возвращает значение типа *Period*.

е. Добавьте классу *Date* операцию *plus* с аргументом *delta* типа *Period*. Результат операции – значение типа *Date*.

4. Класс *MyWindow* уточняет абстрактный базовый класс *Window*. *MyWindow* состоит (композиция) из кнопки класса *Button* и надписи класса *Label*. Отобразите на диаграмме классов.

а. Класс *Label* имеет частный атрибут *text* типа *String* и общедоступную операцию *setText* с параметром *text* типа *String*.

б. Композиция между *MyWindow* и *Button* называется *HoldsButton* . Полнос со стороны кнопки имеет имя *okButton*, защищенную видимость, кратность *1*. Композиция между *MyWindow* и *Label* называется *HoldsLabel* . Украшения полнуса со стороны *Label*: название *textLabel*, частная видимость, кратность *1*.

в. Для реакции на события кнопки реализован паттерн Слушатель (*Listener*) следующим образом. Класс *Button* предоставляет операцию *setClickListener* с единственным параметром *l* типа *IClickListener*. Интерфейс *IClickListener* содержит единственную операцию *onClick* без параметров.

г. Класс *MyWindow* реализует интерфейс *IClickListener* для реакции на нажатие кнопки. Отобразите на диаграмме, что между классом *Button* и *MyWindow* есть ассоциация с именем *NotifyListener* с направлением от кнопки к окну. Укажите, что полнос со стороны окна называется *listener*, имеет тип *IClickListener*, множественную кратность и частную видимость.

д. И *Label* и *Button* имеют строковый атрибут *text*, который можно менять с помощью метода *setText*. Вынесите общий атрибут и метод в абстрактный базовый класс *TextWidget*.

е. Отобразите на диаграмме объектов, как в процессе выполнения объекты связаны между собой: объект *window* класса *MyWindow* связан с кнопкой *button* класса *Button* и с надписью *label* класса *Label*.

5. Интерфейс доступа к коллекции элементов *Collection* обобщает интерфейс работы со списками *List*. Абстрактный класс *BaseCollection* реализует интерфейс *Collection*, абстрактный класс *BaseList* является потомком *BaseCollection* и реализует интерфейс *List*, оставляя операции по хранению данных дочерним классам.

а. Используя наследование, добавьте в модель класс *ArrayList*, реализующий операции со списками с помощью массива.

б. Пусть интерфейс *List* содержит операцию *get* получения элемента списка по заданной позиции *k*. Укажите, в каких классах должна быть объявлена данная операция, чтобы модель была согласованной. Ответ поясните.

в. Пусть интерфейс *Collection* содержит операцию *add* добавления элемента *obj*. Укажите, в пространстве имен каких классов может присутствовать поведение, реализующее операцию *add*. Ответ поясните.

6. Класс *Collections* содержит общедоступную статическую операцию *addAll* с возвращаемым значением типа *boolean*. Первый параметр операции называется *coll* и имеет тип *Collection*, второй параметр называется *elements* и имеет тип *Object* и кратность больше нуля.

а. Добавьте в класс *Collections* статический атрибут *empty* типа *Collection*, предназначенный только для чтения.

б. Реализуйте в классе *Collections* операцию *addAll* с помощью нечеткого поведения (метода), используя операцию добавления элемента *insert (e: Object)* класса *Collection*. Указание. Алгоритм реализации можно показать как псевдокод в комментарии в формате *method = <language> <method body>*.

7. Узел дерева *Node* может иметь несколько дочерних *child* узлов того же класса *Node*.

а. Приведите пример бинарного дерева, состоящего из семи узлов *Node*.

б. Постройте модель дерева, в котором каждый узел имеет от двух до четырех дочерних узлов.

в. Разработайте модель дерева, узлы которого могут быть двух видов: узел *Red* и узел *Black*. *Указание.* Вид узла может изменяться, при этом следует считать, что поведение узла не изменяется при смене типа.

8. У абстрактного класса заказа *Reservation* имеется два потомка: одиночный *Single* и подписка *Subscription*. *Single* связан с одним билетом *Ticket* ассоциацией *reserved*, *Ticket* может быть связан той же ассоциацией не более чем с одним *Single*.

а. Свяжите подписку с билетами в количестве от трех до шести включительно. Билет не обязательно связан с подпиской.

б. Как с помощью ограничений указать, что билет не может быть одновременно связан и с подпиской, и с одиночным заказом?

в. Пусть одиночная подписка наследует свойства одиночного заказа и подписки. С каким максимальным количеством билетов она может быть связана? Ответ поясните.

9. Умный дачный домик *SmartHouse* состоит из четырех стен *Wall* и крыши *Roof*. Домик реагирует на штормовые предупреждения *stormWarning* и укрепляет крышу *harden*, закрывает окна *closeWindows* в стенах. Используемые стройматериалы *Material* характеризуются ценой *price* и удельным весом *unitWeight*.

а. Добавьте стройматериалы для постройки домика: красный и белый кирпич *Brick*, доски *Plank* из сосны и дуба.

б. Укажите, что кирпич является материалом *material* стен. Используя ассоциации, покажите, что каркас крыши *Frame* сделан из не более чем сорока досок и может быть одного из видов *FrameKind*: мансарда, плоский или треугольный.

в. Каркас можно покрыть стройматериалом черепица *Tiling*, отразите это в модели.

г. Допустим, изобретен универсальный стройматериал, заменяющий доски, кирпичи и черепицу. Постройте из него дачный домик. Сколько экземпляров материала понадобится? Ответ поясните.