

Java Base Libraries

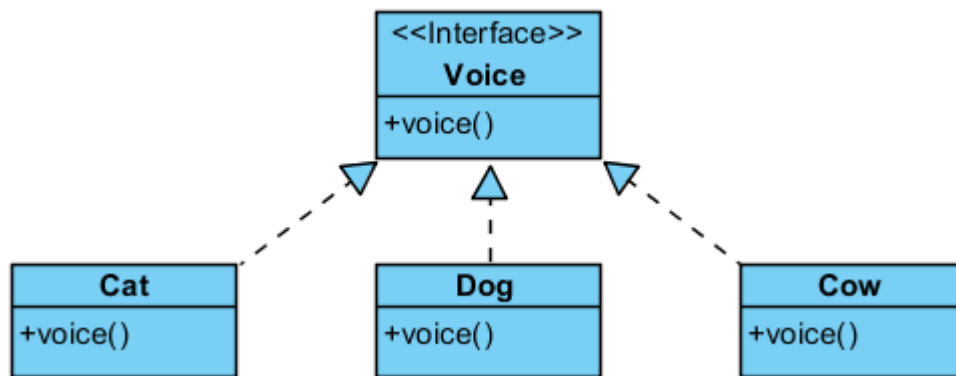
Цель работы:

Сформировать навыки проектирования и реализации интерфейсов Java, закрепить знания в области разработки классов java и научиться переопределять методы `equals()`, `hashCode()`, `toString()`.

Задание. Разработать программу с использованием интерфейсов и переопределить методы Java.

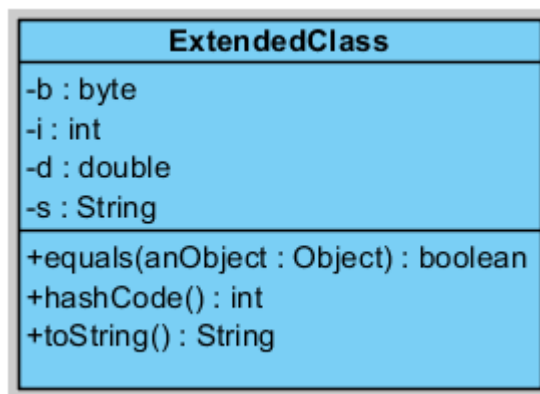
Методические указания

1. Напишите программу, реализующую следующую диаграмму классов:



Переопределяемые методы `voice()` выводят соответствующую строку на консоль.

2. Переработать задачу про игру в кости под использование интерфейсов. Играют N игроков (компьютер в списке последний). Подкидываются одновременно K кубиков. Выигрывает тот, у кого большая сумма очков. Кто выиграл, тот и кидает первым в следующем кону. Игра идет до 7 выигрышей. Начинаете игру Вы.
3. Напишите программу, реализующую изображенный класс:



4. Выполните задание по варианту. Выбор осуществляется по первой букве фамилии.

Классы, реализуемые в задании, не имеют метода `main()` и перед использованием должны быть инстанцированы (должен быть создан объект класса) в методе `main()` основного класса.

№ варианта	Задание
А	Создайте интерфейс <code>UsdRubConverter</code> с методами <code>setRate()</code> , <code>convertRubToUsd()</code> и <code>convertUsdToRub()</code> . Реализуйте интерфейс в классе <code>UsdRubConverterImpl</code> . Метод <code>setRate(double rate)</code> устанавливает курс доллара к рублю (изначально 60) равным <code>rate</code> , методы <code>convert</code> принимают на входе значения в рублях/долларах и возвращают конвертированное в доллары/рубли значение (использовать тип <code>double</code>).
Б	Создайте интерфейс <code>Cat</code> с методами <code>feed()</code> и <code>isHungry()</code> . Реализуйте интерфейс в классе <code>CatImpl</code> . Метод <code>isHungry()</code> должен выводить в консоль "I'm hungry, bro!", если до него не было ни одного вызова метода <code>feed()</code> , и "I just want to sleep for a bit, OK?" в противном случае.
В	Создайте интерфейс <code>Sleepyc</code> методами <code>sleep()</code> , <code>wakeUp()</code> и <code>ask()</code> . Реализуйте интерфейс в классе <code>SleepyImpl</code> . Метод <code>sleep()</code> устанавливает флаг <code>awake</code> в <code>false</code> , метод <code>wakeUp()</code> в <code>true</code> . Метод <code>ask()</code> печатает в консоль "BOO!", если флаг установлен в <code>true</code> , и "zzz..." в противном случае.
Г	Создайте интерфейс <code>Dice</code> с методами <code>roll()</code> и <code>look()</code> . Реализуйте интерфейс в классе <code>DiceImpl</code> . Метод <code>roll()</code> генерирует случайное число от 1 до 6, метод <code>look()</code> отображает последнее сгенерированное число в консоли.
Д	Создайте интерфейс <code>Altitude</code> с методами <code>higher()</code> , <code>lower()</code> и <code>howHigh()</code> . Реализуйте интерфейс в классе <code>AltitudeImpl</code> . Метод <code>higher()</code> прибавляет к текущей высоте (изначально высота равна нулю) единицу. Метод <code>lower()</code> вычитает из текущей высоты

	единицу. Метод howHigh() печатает в консоль “Flying x meters above the ground”, если $x > 0$, “Swimming x meters below the ground”, если $x < 0$. В оставшемся случае метод печатает “Standing on the ground!”.
Е, Ё	Создайте интерфейс Oddity с методами setNumber(int x) и isOdd(). Реализуйте интерфейс в классе OddImpl. Метод setNumber(int x) устанавливает текущее число (изначально равное нулю) в x. Метод isOdd() выводит в консоль «число нечётно» либо «число чётно» в зависимости от четности установленного числа.
Ж, З	Создайте интерфейс ThreeNumbers с методами setX(int value), setY(int value), setZ(int value), getX(), getY(), getZ(), getLessThan(int value). Реализуйте интерфейс в классе ThreeNumbersImpl. Опишите в нём поля x, y и z, а также сеттеры и геттеры для них. Метод printLessThan(int value) печатает в консоль значения переменных x, y, z, меньших value.
И, Й	Создайте интерфейс RNG с методами generate() и anyLuck(). Реализуйте интерфейс в классе RNGImpl. Метод generate() генерирует случайное число от 1 до 9. Метод anyLuck() печатает в консоль «Lucky!», если последнее сгенерированное значение равно 7, иначе печатает “No luck...”
К	Создайте интерфейс Runner с методами run() и getDistance(). Реализуйте интерфейс в классе RunnerImpl. Метод run() должен прибавлять к текущей дистанции (изначально равной нулю) число 1 при первом вызове, число 2 при втором, число 3 при третьем и т.д. Метод getDistance() должен печатать текущую дистанцию на экран.
Л	Создайте интерфейс ThePriceIsRight с методами setPrice(int price) и checkPrice(int price). Реализуйте интерфейс в классе ThePriceIsRightImpl. Метод setPrice(int price) устанавливает текущую цену (изначально 1000) в price. Метод checkPrice(int price) выводит в консоль «lower», если текущая цена меньше price, «higher», если текущая цена больше price, и "The price is right!", если текущая цена в точности равна price.
М	Создайте интерфейс Magic с методами setNumber(int x), doMagic(), int getNumber(). Реализуйте интерфейс в классе MagicImpl. Метод setNumber(int x) устанавливает текущее число в x. int getNumber() возвращает текущее число. doMagic() переставляет цифры числа в обратном порядке.
Н, О	Создайте интерфейс Patience с методом hey(). Реализуйте интерфейс в классе PatienceImpl. Метод hey() должен печатать в консоль “Hey there!” первые три раза, которые он вызван. Во все последующие вызовы метод печатает в консоль “You can stop it now...”
П	Создайте интерфейс Black с методами setColor(String color) и

	isBlack(). Реализуйте интерфейс в классе BlackImpl. Метод setColor(String color) должен устанавливать текущий цвет в color. Метод isBlack() должен печатать в консоль “It is black”, если текущий цвет == “black”, и “it isn’t black” в противном случае.
Р	Создайте интерфейс MyString с методами addLetter() и show(). Реализуйте интерфейс в классе MyStringImpl. Метод addLetter() добавляет в конец текущей строки (изначально текущая строка пуста) символ ‘a’ или ‘b’ (случайно). Метод show() печатает в текущую строку в консоль.
С	Создайте интерфейс TwoNumbers с методами setX(), setY(), getDifference(). Реализуйте интерфейс в классе TheNumbersImpl. setX() должен устанавливать значение поля x, setY() – значение поля y. getDifference() печатает на экран абсолютное значение разности x и y.
Т	Создайте интерфейс PointAndSegment с методами setPointX(int value), setX1(int value), setX2(int value), isInside(). Реализуйте интерфейс в классе PointAndSegmentImpl. В нём должны присутствовать поля x1, x2 – координаты начала и конца отрезка на прямой Ох, и pointX – координата точки. Методы set присваивают соответствующим переменным значение value. Метод isInside() проверяет, находится ли точка внутри отрезка, и печатает результат проверки в консоль
У, Ф	Создайте интерфейс RestrictedNumber с методами setValue(int value), getValue(). Реализуйте интерфейс в классе RestrictedNumberImpl. Класс должен содержать поле x. setValue(int value) устанавливает полю x значение value. При этом x не должен становиться > 50 или < -50. Если же это происходит, x следует приравнять к 50, либо -50. Метод getValue() печатает значение x в консоль.
Х, Ц, Ч	Создайте интерфейс Palindrome с методами setString(String value), isPalindrome(). Реализуйте интерфейс в классе PalindromeImpl. Метод setString(String value) устанавливает значение текущей строки равным value. Метод isPalindrome() выводит в консоль «Строка является палиндромом», если текущая строка им является, либо «Строка не является палиндромом» в противном случае. Палиндром – строка, читающаяся одинаково как слева направо, так и справа налево.
Ш, Щ, Ы	Создайте интерфейс ThreeNumbers с методами setX(int value), setY(int value), setZ(int value), getX(), getY(), getZ(), sort(). Реализуйте интерфейс в классе ThreeNumbersImpl. В нём должны присутствовать поля x, y и z, а также сеттеры и геттеры для них. Метод sort() упорядочивает значения по неубыванию (x – наименьшее, y – среднее, z – наибольшее).
Э, Ю, Я	Создайте интерфейс Accumulate с методами add(int x) и show().

	Реализуйте интерфейс в классе AccumulateImpl. Метод add(int x) должен добавлять к текущему числу (изначально равному 0) число x, метод show() выводить текущее число в консоль.
--	---

Предоставьте отчет, который содержит: текст программного кода выполненных заданий, скрин результатов работы и ссылку на ваш проект на GitHub или любой другой веб-сервис для хостинга IT-проектов. Все программы оформлять в соответствии с Code Convention, при оформлении кода использование комментариев обязательно.