Для всех задач необходимо дополнительно создавать класс Application, который будет содержать статический метод main.

**Задача 1**

Дано: Существуют 2 массива целых чисел dividends и dividers размерностями m и n соответственно, числа m и n – случайные числа от 5 до 10.

Задание:

Заполнить данные массивы случайными числами: массив dividends в диапазоне от 4 до 9, массив dividers случайными числами от 0 до 3.

Создать двумерный массив целых числе result[n][m]. Результатами матрицы является m строк и n столбцов, где каждая i-ая строка является результатом деления элемента dividends[i] на каждый элемент массива dividers. В случае деления на ноль значение результата должно быть равно -1. Использовать конструкцию try/catch, с отлавливанием конкретного возникающего исключения.

Вывести на экран dividend, divider, result

|  |  |  |
| --- | --- | --- |
| Входные данные dividend[] | Входные данные divider[] | Полученный результат |
| [x,  y,  z] | [i,  j,  k] | [[x/i, x/j, x/k],  [y/i, y/j, y/k],  [z/i, z/j, z/k]] |
| [9,  8,  7,  6,  5] | [0,  1,  2,  3] | [[ -1, 9, 4, 3 ],  [-1, 8, 4, 2],  [-1, 7, 3, 2],  [-1, 6, 3, 2],  [-1, 5, 2, 1]] |

**Задача 2**

Реализовать класс Cat, содержащий поля String name, int age. Конструктор класса должен принимать оба поля в качестве параметров.

В методе main создать массив Cat[5], который заполнить следующими данными: [cat0, cat1, null, cat3, cat4]. В цикле выводить на экран информацию об элементах в формате

Имя: %s, Возраст: %d.

Возникающее исключение должно быть корректно обработано, на экран должно быть выведено соответствующее сообщение. Информация об объектах, которые находятся в массиве после элемента null, выводиться не должна

**Задача 3**

Дано: На итерации осуществляется деление числа m на число n. Числа m и n выбираются случайно в диапазоне [0..4].

Задание: Для числа итераций total, выбираемого случайно из диапазона [500..1000] посчитать количество операций, в процессе которых было произведено деление на ноль. Вывести на экран сообщение в формате:

Из %d операций деления %d было выполнено с ошибкой

**Задача 4**

Реализовать класс Cat, содержащий поля String name, int age. Конструктор класса должен принимать оба поля в качестве параметров.

Создать классы исключений AgeUnderZeroException и AgeTooHighException. Первое сделать непроверяемым, а второе – проверяемым.

Исключение AgeUnderZeroException должно возникать, если в конструктор передано отрицательное значение age. Исключение AgeTooHighException должно возникать, если в конструкто передано значение age > 20.

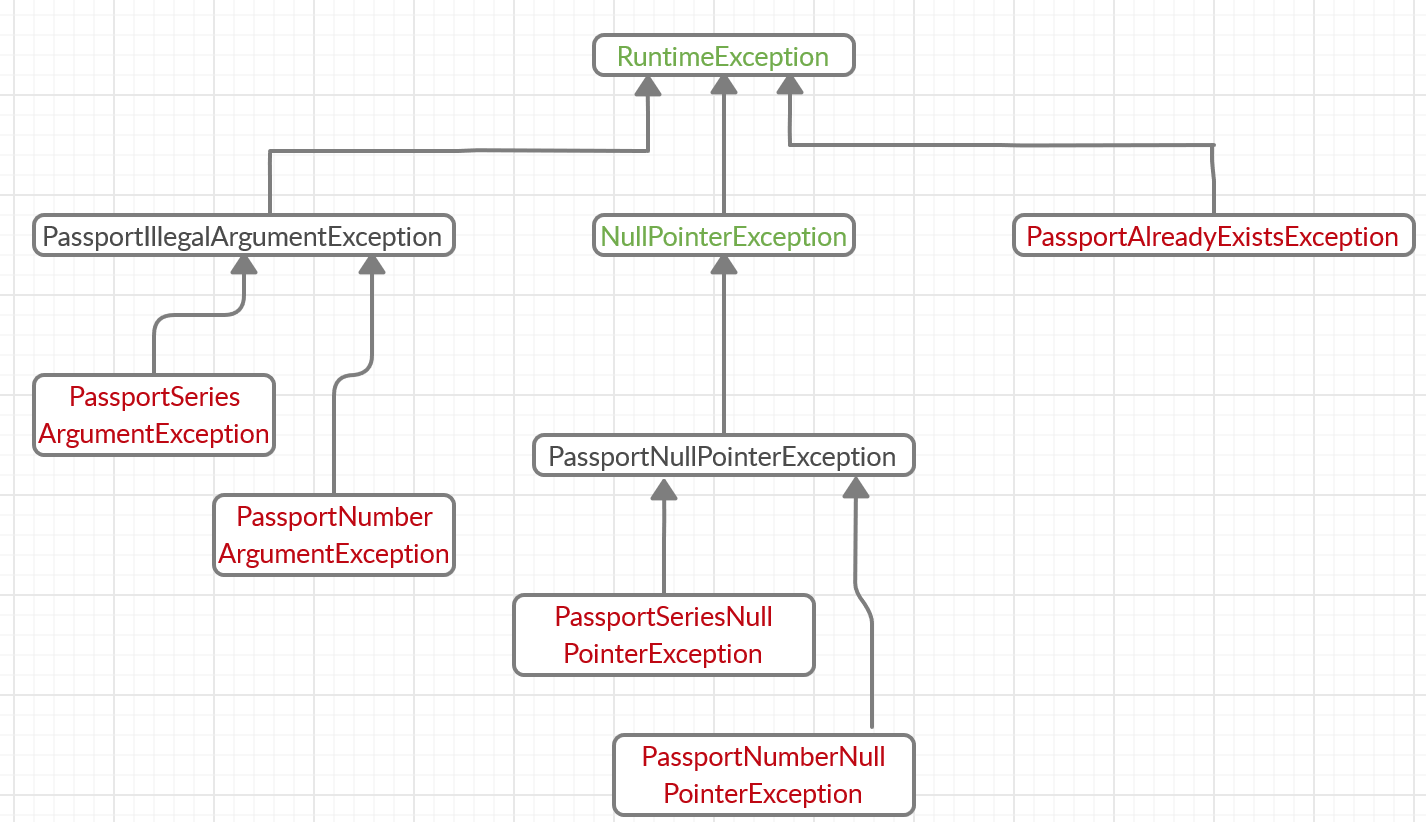
В конструкторе необходимо обработать исключение AgeUnderZeroException – вывести только первую строку из StackTrace. Исключение AgeTooHighException необходимо пробросить наверх и обработать в методе main – вывести сообщение «Слишком большой возраст для кота»

В методе main необходимо создать массив Cat[10]. Создать цикл, который будет заполнять данный массив. Возраст выбирать случайно из диапазона [-50..50]. В случае возникновения исключения элемент массива заполнить значением null. Вывести данные из массива в формате «Имя: %s, Возраст: %d». В случае возникновения исключения вывести «Ошибка вывода данных».

**Задача 5**

Реализовать класс Passport, содержащий поля String series, String number. Конструктор класса Passport должен принимать два параметра и выбрасывать соответствующие исключения, если возникают соответствующие условия. Если исключений не возникло, конструктор должен устанавливать поля объекта переданными значениями. Из конструктора должны пробрасываться наверх 3 исключения: PassportIllegalArgumentException, NullPointerException, PassportAlreadyExistsException.

Реализовать иерархию исключений согласно схеме ниже.



Реализовать логику работы исключений следующим образом:

|  |  |  |
| --- | --- | --- |
| Исключение | Условия, его вызывающие | Обработка исключения |
| PassportSeriesArgumentException | Неверный формат серии паспорта. Должен состоять из 2 цифр, пробела, 2 цифр. Например: 12 34 | Вывести в консоль сообщение  «Введите верные значения серии и номера паспорта» |
| PassportNumberArgumentException | Неверный формат номера паспорта. Должен состоять из 6 цифр. Например: 123456 | Вывести в консоль сообщение  «Введите верные значения серии и номера паспорта» |
| PassportSeriesNullPointerException | В конструктор в качестве серии паспорта передан null | Вывести в консоль сообщение  «Введите верные значения серии и номера паспорта» |
| PassportNumberNullPointerException | В конструктор в качестве номера паспорта передан null | Вывести в консоль сообщение  «Введите верные значения серии и номера паспорта» |
| PassportAlreadyExistsException | В конструктор в качестве серии и номера паспорта переданы значения, объект с которыми был создан ранее | Вывести в консоль сообщение  «Данные серия и номер паспорта уже существуют» |

При обработке исключений отлавливать необходимо конкретные исключения без обобщений (данные исключения выделены красным шрифтом на схеме). В классе Application создать метод Passport createPassport(String series, String number), который будет возвращать объект типа Passport. В случае исключения данный метод должен возвращать null.