**Практическое задание**

**Реализуйте консольное приложение “Блокнотик”.**

Приложение должно позволять создавать один блокнот, добавлять в него записи, искать запись по дате или с требуемым содержимым, удалять запись. Также разработайте тесты для проверки корректности работы классов приложения.

*Примечания:*

При реализации приложения старайтесь придерживаться правил layered architecture, помните и про декомпозицию ваших локальных решений.

Вид(работу с клиентом) в приложении (меню и другое интерактивное общение с пользователем) реализовывать в консоли.

*Отчетность:*

Реализованную задачу необходимо разместить на github, ссылку на репозиторий выслать на почту [Olga\_Smolyakova@epam.com](mailto:Olga_Smolyakova@epam.com) не позже 31.12.22.

**Теоретическое задание**

Письменно ответьте на вопросы. Ответы запишите в файле Microsoft Word (Open Office). Не списываете и не делайте прямой copy-paste из источников, пишите свои ответы своими словами (ну пожааалуйстаааа!!!).

1. Расскажите, как вы понимаете, что такое класс и объект? Зачем в приложении использовать классы и объекты? Как в Java они создаются? Приведите как можно более подробное описание элементов, которые можно встретить при разработке класса (например конструкторов и пр.)

Ответ: Класс – это «чертеж» программы. Здесь мы указываем, какие данные будут использоваться, каких типов будут эти данные и как с этими данными работать.

Любой объект может обладать двумя основными характеристиками: состоянием и поведением. Поведение описывается исключительно внутри классов и называется «методом».

Для хранения состояния объекта в классе применяются поля или переменные класса.

Объект – построенная по образу и подобию «класса» сущность.

Для объявления класса служит ключевое слово class.

Новый объект создаётся из существующего класса при помощи ключевого слова new.

Кроме обычных методов в классах есть специальные методы, которые называются конструкторами. Они нужны при создании нового объекта данного класса.

Если в классе не определено ни одного конструктора, то для этого класса автоматически создается конструктор без параметров.

1. Расскажите как можно подробнее про передачу параметров примитивных и ссылочных типов в методы. Приведите примеры.

Ответ: В Java передача параметров примитивных типов происходит «по значению». Это значит, что одна переменная не записывается в другую, а значение копируется и присваивается новой переменной.

Для ссылочных типов данных происходит передача «по ссылке», т.е. сам объект никогда не передается, передается лишь копия ссылки на этот объект.

1. Объясните, как вы понимаете, что такое исключение в Java программе, зачем они используются? Приведите иерархию исключений.

Исключение – это ошибка во время выполнения программы. В случае его возникновения прекращается нормальное выполнение программы, и она завершается в аварийном режиме. Такие случаи требуют специальной обработки. Все исключения являются объектами. Они делятся на несколько классов, но все они имеют общего предка – класс Throwable. Его потомками являются классы Exception и Error. Все Exceptions делятся на два типа: контролируемые (checked) и неконтролируемые (unchecked), к которым относятся ошибки и Runtime Exceptions.

Контролируемые исключения происходят на этапе компиляции и требуют разрешения.

Неконтролируемые происходят во время выполнения. Эти исключения включают в себя: погрешности программирования, логические ошибки или неверное использование API.

Ошибки не являются исключениями, это проблемы, которые возникают независимо от пользователя либо программы.

1. Что такое коллекция? Приведите основные методы работы сo знакомыми вам классами-коллекциями.

Коллекция – это объект, способный хранить группу одинаковых элементов. Она содержит методы для операций с однородными данными. Коллекции могут хранить любые ссылочные типы данных. Есть два основных интерфейса, от которых наследуются все остальные классы коллекций: Collection и Map.

Collection хранит объекты в ячейках (как в массивах). С ними возможны все манипуляции: удаление, вставка нового, поиск, возврат размера, замена и другие.

Map – хранит данные в виде пары «ключ - значение».

1. Объясните, что такое перегрузка и переопределение методов? Как вы думаете, когда их следует использовать.

Переопределение методов позволяет подклассу обеспечить специфическую реализацию метода, который уже реализован в суперклассе. У переопределенного метода должен быть тот же тип возвращаемого значения, что и у метода родителя (либо можно изменить его на производный от используемого в методе базового класса). Для переопределения метода в производном классе необходимо повторить объявление метода из базового класса и дать ему новую реализацию. При определении можно расширить область его видимости на более широкий. С помощью этого механизма достигается поведение, когда экземпляр хранится под типом родителя, но реализация методов используется специфичная для этого конкретного подтипа.

Перегрузка же методов – это возможность создавать несколько методов с одинаковым названием, но разными параметрами и возвращаемыми значениями. По сути – это разные методы, просто с одинаковым названием.