

Чем можно пользоваться во время выполнения:

- <https://docs.oracle.com/en/java/javase/11/docs/api/>
- <https://developer.android.com/reference/>

1. Реализовать Java-программу в Apache NetBeans IDE, которая преобразовывает введенную пользователем с клавиатуры строку:

- разворот строки;
- удаление первого и последнего символа в строке;
- перевод строки в верхний регистр.

Три новых преобразованных строки необходимо сохранить в текстовый файл.

2. Реализовать Java-программу в Apache NetBeans IDE, которая вычисляет максимальное и минимальное число из введенных пользователем с клавиатуры числовых значений через запятую. Результаты отображаются в консоли. В решении необходимо использовать цикл while. Проверки на некорректные значения – обязательны!

3. Реализовать Java-программу в Apache NetBeans IDE, выполняющую вывод в консоль отсортированных по имени сущностей типа Person (объекты хранятся в ArrayList). Помимо имени, класс Person должен содержать еще 3 переменных класса. В решении необходимо использовать интерфейс Comparable. Заполнение коллекции выполняется пользователем с клавиатуры во время выполнения программы (минимум 3 элемента коллекции).

4. Реализовать Java-программу в Apache NetBeans IDE, выполняющую вывод в консоль отсортированного по значению ключа HashMap (ключ – типа String). Заполнение HashMap выполняется пользователем с клавиатуры во время выполнения программы (минимум 5 значений). Необходимо предусмотреть проверки на некорректные значения и обработку исключений.

5. Реализовать Java-программу в Apache NetBeans IDE, которая загружает содержимое текстового файла, удаляет все гласные буквы и выводит полученное значение в консоль. Путь к файлу вводит пользователь с клавиатуры. Проверки на некорректные значения – обязательны!

6. Реализовать Java-программу в Apache NetBeans IDE, которая сериализует в файл экземпляр класса сущности Message (класс должен содержать минимум 3 переменных). Путь к файлу вводит пользователь с клавиатуры.

7. Реализовать Java-программу в Apache NetBeans IDE, которая вычисляет площадь прямоугольника в отдельном потоке и выводит результат в консоль. Создание потока должно быть реализовано с использованием интерфейса Runnable. Значения сторон вводит пользователь с клавиатуры. Проверки на некорректные значения – обязательны!

8. Реализовать Java-программу в Apache NetBeans IDE, которая вычисляет в отдельном потоке факториал введенного пользователем числа с клавиатуры и выводит полученное значение в консоль. Необходимо использовать цикл while. Создание потока должно быть реализовано с использованием наследника класса Thread. Проверки на некорректные значения – обязательны!

9. Дан исходный код Java-программы. При запуске программы (а также в ходе работы), возникают ошибки. Провести отладку программы (debugging), выявить в каких участках программы возникают ошибки, исправить их, запустить рабочий вариант. В данном задании проверяются знания темы обработки исключений, а также умение работы с отладчиком Apache NetBeans IDE.

Загрузите программу по адресу: <https://goo.su/MAN2U>

Программа считывает число с клавиатуры, проводит математические операции и отправляет некоторое значение на сервер. Сервер возвращает измененное число. Приложение содержит ошибки компиляции, также в приложении могут возникнуть ошибки при неправильной эксплуатации. Задача исправить ошибки и предусмотреть обработку ошибок в «уязвимых» участках кода.

10. Реализовать Java-программу в Apache NetBeans IDE, которая сохраняет в файл содержимое веб-страницы, адрес которой вводит пользователь с клавиатуры. Использование многопоточности – обязательно.

11. Реализовать мобильное приложение в Android Studio, которое складывает два целых числа, введенных пользователем в поля ввода (EditText). Результат вычисления необходимо отобразить в виджете TextView после нажатия на кнопку (Button) и отобразить в лог. В макете activity разрешено использовать контейнеры только одного типа: LinearLayout. В приложении необходимо использовать Library Data Binding для привязки виджетов. Проверки на некорректные значения – обязательны!
12. Реализовать мобильное приложение в Android Studio, состоящее из двух activity, таким образом, чтобы в первую activity вернулся результат математической операции из второй activity. Подробнее: Вторая activity запускается по нажатию на кнопку (Button) в первой activity. Во второй activity пользователь вводит числовое значение в EditText. Далее это значение возводится в степень 2. Пользователь нажимает на кнопку (Button) и приложение возвращается в первую activity и отображает в поле TextView вычисленное значение. Проверки на некорректные значения – обязательны!
13. Реализовать мобильное приложение в Android Studio, с помощью которого по нажатию на картинку (ImageView) можно поделиться текстом из виджета EditText через другое приложение (мессенджеры, смс и т.п.). В приложении необходимо использовать Library Data Binding для привязки виджетов. Картинка для imageView: <https://goo.su/TpgLD9l>.
14. Реализовать мобильное приложение в Android Studio, в котором по нажатию на кнопку (Button) будет заменяться содержимое экрана, расположенного ниже кнопки. При выполнении задания обязательно использовать фрагменты (динамическая регистрация). Содержимое экранов может быть любое.
15. Реализовать мобильное приложение в Android Studio, в котором при нажатии на кнопку (Button) создается база данных SQLite с таблицей для хранения списка сообщений (Message). Таблица должна содержать минимум 4 столбца: идентификатор сообщения (число), тело сообщения (строка), дата и время отправки сообщения (строка), флаг является ли сообщение прочитанным или нет.

16. Реализовать мобильное приложение в Android Studio, в котором значение (тип String), записанное в поле виджета EditText, сохраняется в хранилище SharedPreferences при нажатии на кнопку (Button). При нажатии на вторую кнопку, значение загружается из SharedPreferences и отображается в TextView и в лог.
17. Реализовать мобильное приложение в Android Studio, в котором значение, записанное в поле виджета EditText, сохраняется в текстовый файл во внутренней памяти устройства при нажатии на кнопку (Button). Имя файла задает пользователь (вводит во второй EditText). При сохранении файла необходимо использовать многопоточность (отдельный поток).
18. Реализовать мобильное приложение в Android Studio, в котором по нажатию на кнопку (Button) «Старт» будет запускаться фоновый сервис. В сервисе должен запускаться новый поток через WorkManager (код в отдельном потоке может быть любым). По нажатию на кнопку «Стоп» сервис должен быть остановлен, а задача WorkManager – отменена.
19. Реализовать мобильное приложение в Android Studio, которое загружает содержимое интернет-страницы в лог по нажатию на кнопку (Button). Адрес интернет страницы вводит пользователь в EditText. Допускается использовать любой инструмент для работы с сетью.
20. Реализовать мобильное приложение в Android Studio, которое загружает на сервер с помощью POST-запроса протокола HTTP текстовое сообщение. Текстовое сообщение вводит пользователь в поле виджета EditText. По нажатию на кнопку (Button) выполняется POST-запрос. Сообщение передается в поле «text» тела запроса. Ответ сервера необходимо отобразить в консоль. Допускается использовать любой инструмент для работы с сетью (адрес сервера и url запроса необходимо получить у преподавателя).