

Задача: есть ПК, к нему подключены 4 монитора. В памяти ПК сохранен каталог с 4 изображениями. Написать программу, выводящую на монитор соответствующие по счету изображение.

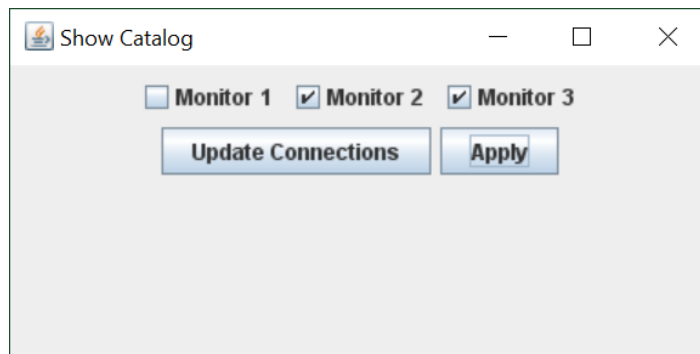
Решение:

1. Был выбран ЯП Java, т.к. он наиболее привычен, и обладает инструментами для решения этой задачи.

2. В среде разработки IntelliJ IDEA 2022.1.2 был создан проект. В главном (и единственном) классе был написан код, представленный в конце этого документа.

3. Логика программы:

При запуске программы на главном мониторе открывается окно (JFrame), на нем отображается список подключенных мониторов с чекбоксами напротив, и две кнопки: “Обновить подключения” (“Update Connections”) и “Применить” (“Apply”).



Информацию о кол-ве подключенных мониторов программа получает с помощью строки

```
GraphicsDevice[] devices =
```

```
GraphicsEnvironment.getLocalGraphicsEnvironment().getScreenDevices();
```

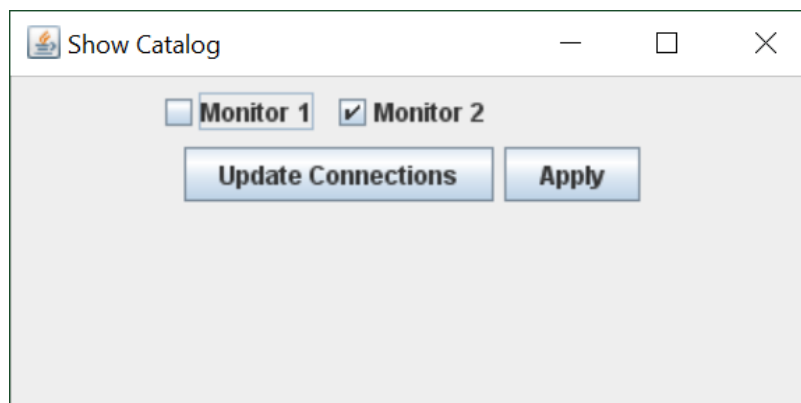
Класс *GraphicsDevice* описывает графические устройства.

Класс описывает коллекцию объектов *GraphicsDevice*, которые могут быть экранами, принтерами или буферами изображений и являются адресатами методов рисования *Graphics2D*.

getLocalGraphicsEnvironment() - Возвращает локальную графическую среду.

getScreenDevices() - Возвращает массив всех экранных устройств.

При отключении одного из мониторов, и нажатии на кнопку “Обновить подключения”, окно будет выглядеть так:



При нажатии на кнопку “Применить” программа откроет окна с изображениями на тех мониторах, напротив которых стоит галочка в окне “Show Catalog”.

Расчет того, какое по счету изображение из каталога будет выводиться на следующий монитор осуществляется с помощью:

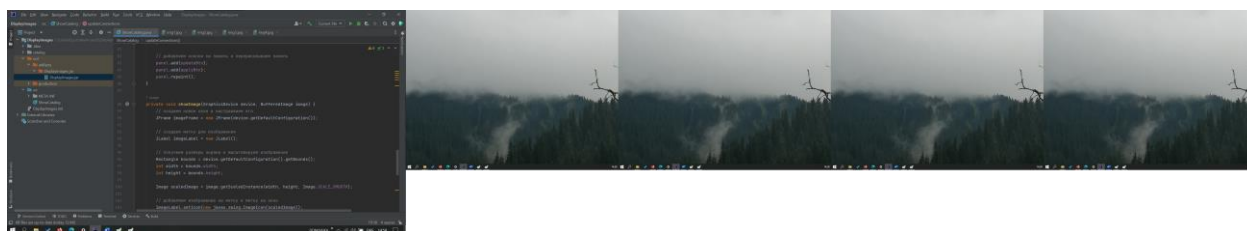
```
imageIndex = (imageIndex + 1) % images.length;
```

Например, если изображений в каталоге 4, то на 7-ой подключенный монитор, будет выведено 3-е изображение (при условии, что поставлены все галочки).

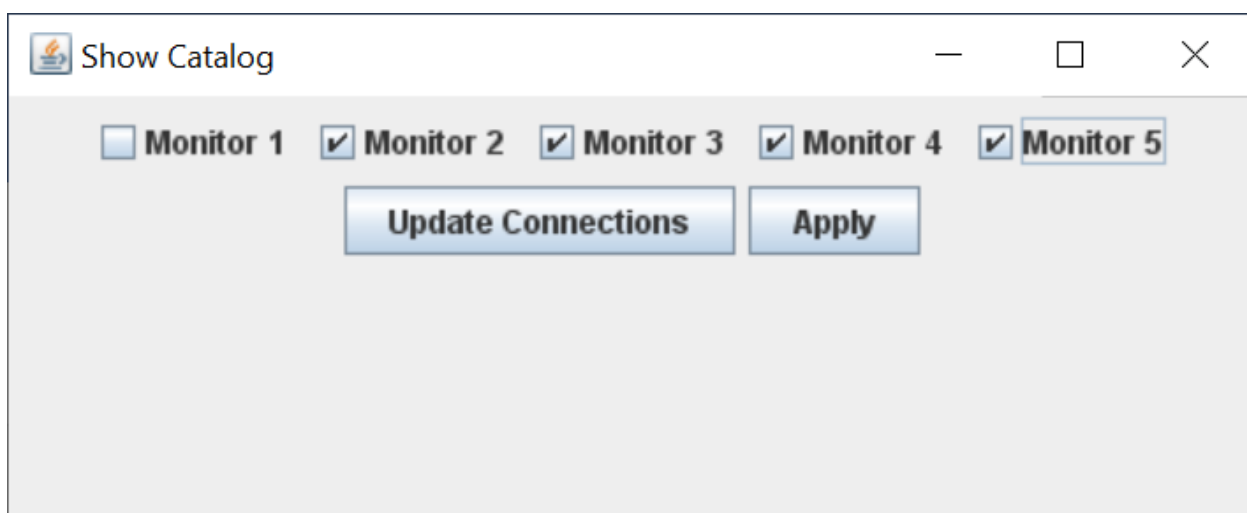
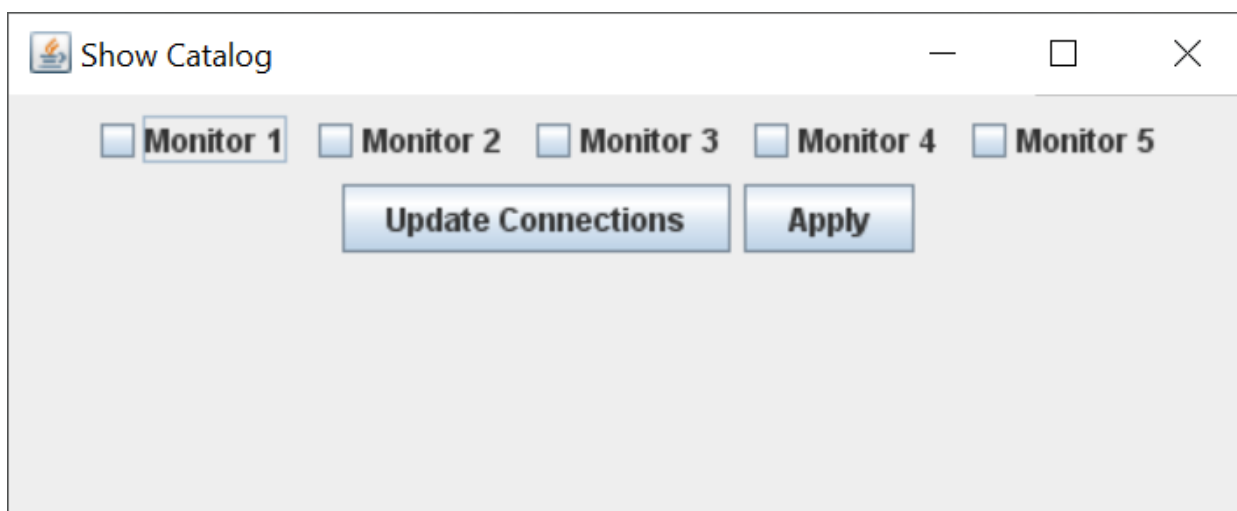
4. Тестирование работы программы.

Можно подключать мониторы разными способами, например с помощью HDMI. Во время тестирования у меня не оказалось с собой кабеля, хотя я и проверял работу программы с этим подключением. Способ подключения не влияет на работу программы.

4.1 Было подключено 4 дополнительных монитора (4 телефона с помощью драйвера spacedesk):



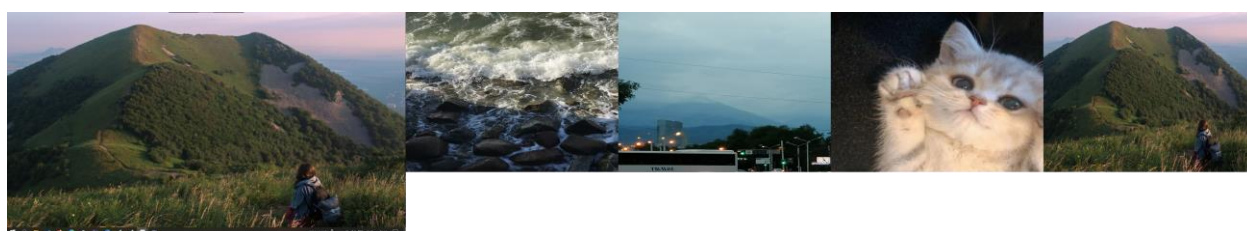
4.2 Далее было запущено созданное приложение и выбрано 4 монитора:



4.3 После нажатия на кнопку “Apply” на мониторы были выведены изображения из каталога:

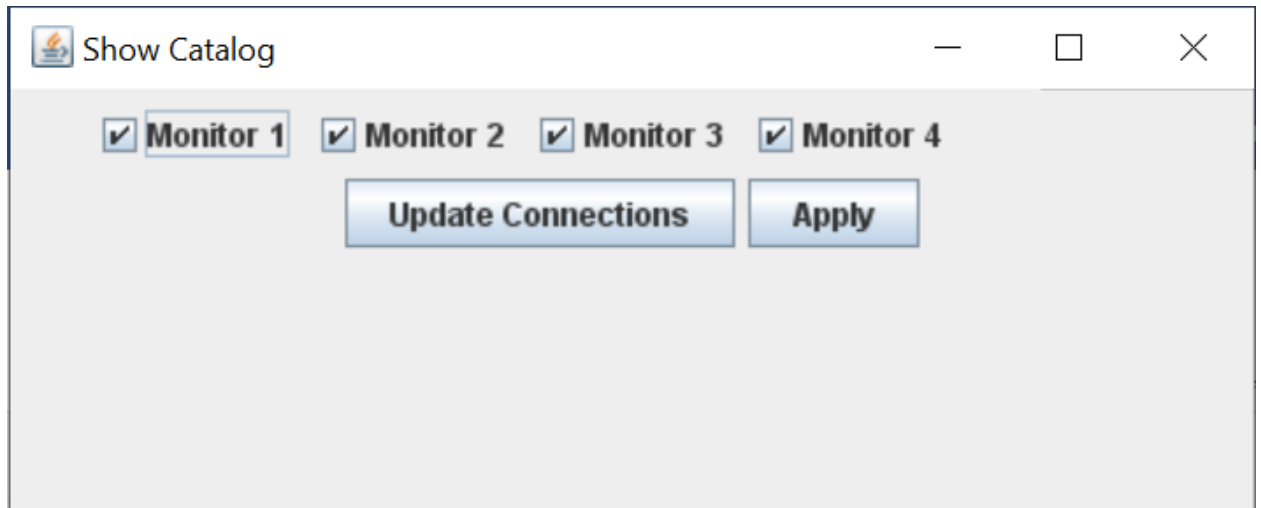


Была добавлена галочка напротив первого монитора и нажата кнопка “Apply”:

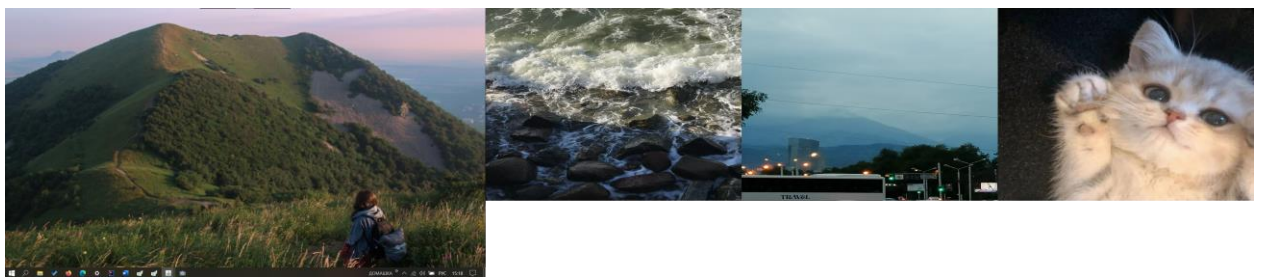


По рисунку выше видно, что на пятом мониторе появилось первое изображение.

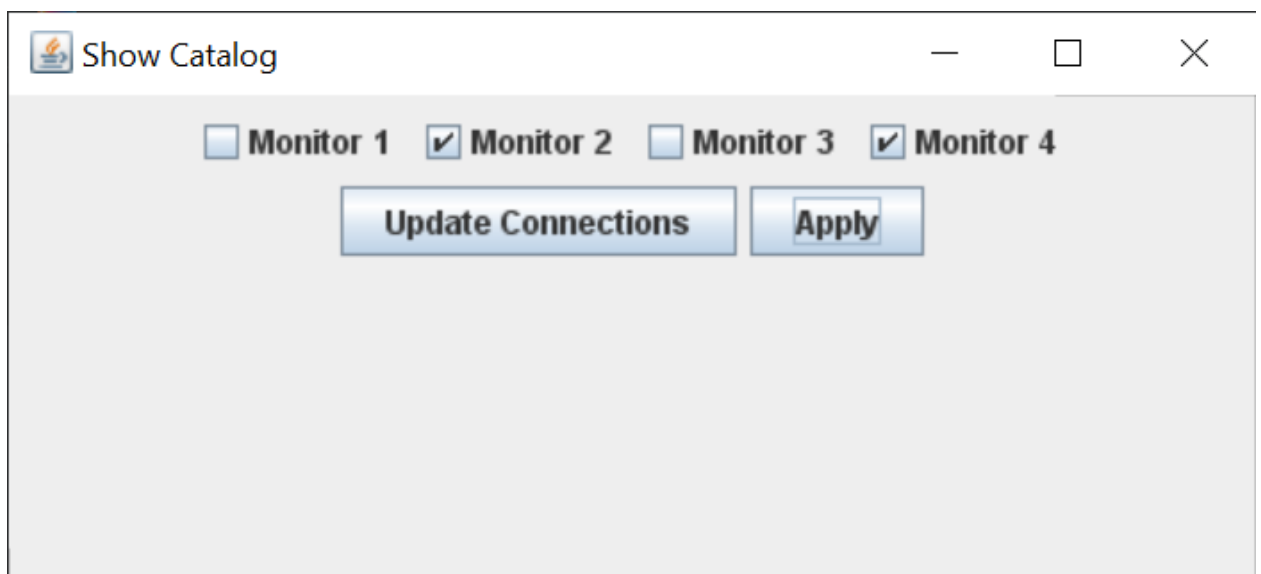
4.4 Монитор 3 был отключен. Далее была нажата кнопка “Обновить подключения”:

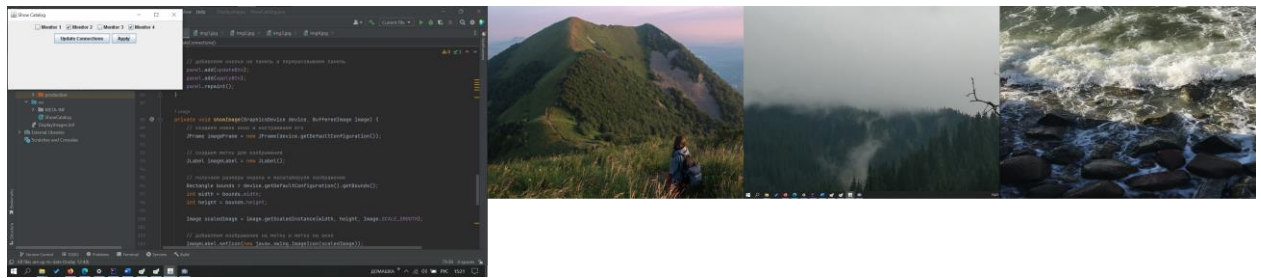


Нажата кнопка “Применить”:



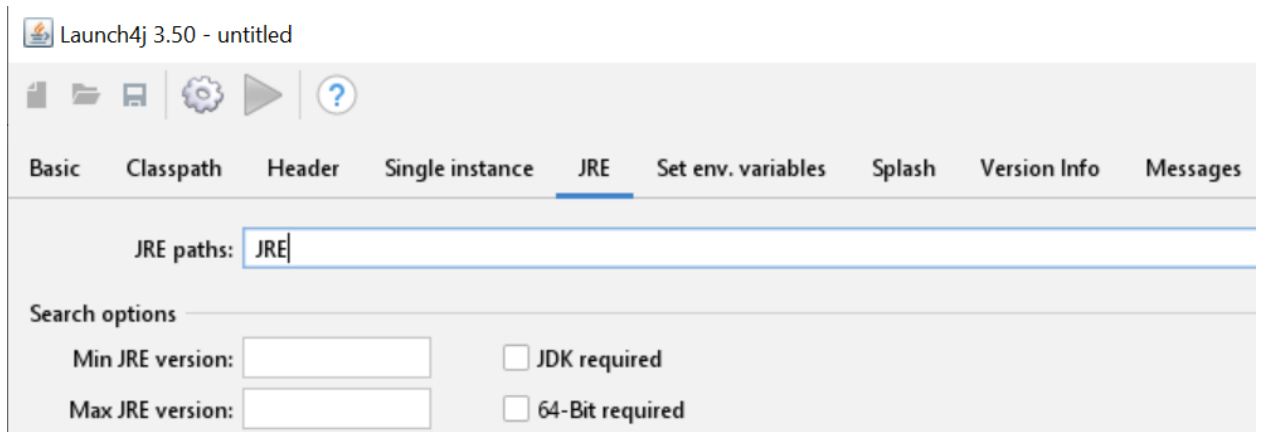
4.5 Были поставлены галочки напротив второго и четвертого монитора и нажата кнопка “Применить”.



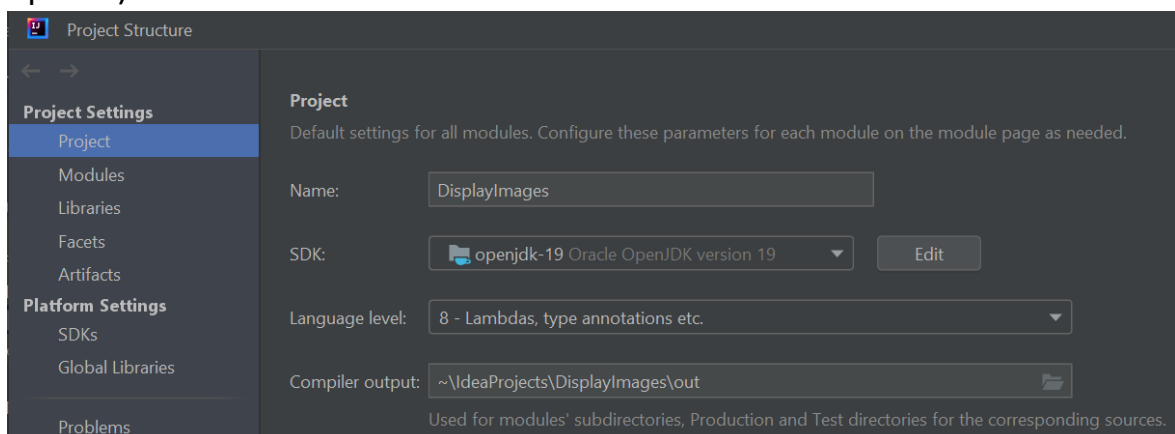


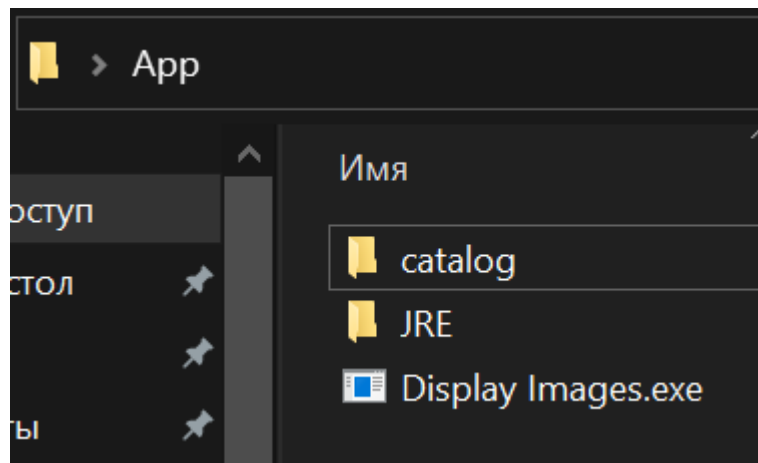
На рисунке выше видно, что на втором и четвертом мониторе открылось окно с изображением, а на первом и третьем показан рабочий стол или окно с программой.

5. Создан .exe файл программы. Для создания exe файла был создан jar файл проекта. Далее с помощью программы launch4j из jar был сформирован exe файл. При создании нужно учесть, что java машина стоит не на всех компьютерах, и для автономности во вкладке JRE можно указать, где exe файл должен найти дистрибутив.



После создания exe файла помещаем его в директорию вместе с папкой JRE, и каталогом изображений. В папку JRE было утсановлено "jre-8u341-windows-x64". (Версия JRE в папке должна быть не ниже той, на которой собирался проект)





Если же на Ваш компьютер установлена Java, то можно обойтись jar файлом, помещенным в папку вместе с каталогом.

