

Лабораторная работа № 2

Тема работы:

Группирование потоков. Приоритеты потоков.

Теоретическая часть:

Группирование потоков

Группа потоков представляет собой набор потоков. Кроме того, в группу потоков могут входить и другие группы потоков. Группы потоков образуют дерево, в котором каждая группа потоков, кроме начальной, имеет родителя.

Потоку разрешен доступ к информации о своей собственной группе потоков, но запрещен доступ к информации о родительской группе потоков или других группах потоков.

Переведено с помощью www.DeepL.com/Translator (бесплатная версия)

Источник: <https://docs.oracle.com/javase/8/docs/api/java/lang/ThreadGroup.html>

```
ThreadGroup some_group = new ThreadGroup();  
Thread thread_in_group = new Thread(some_group);
```

В фрагменте кода, который показан выше, создается группа для потоков и потом создается поток, который сразу добавляется в группу путем указания группы в аргументах конструктора.

```
ThreadGroup another_group = new ThreadGroup(some_group);
```

Таким же образом как с добавлением потока в группу, можно у добавить другую группу в первоначальную группу.

Приоритеты потоков

Как мы уже знаем, Java, будучи полностью объектно-ориентированной, работает в многопоточной среде, в которой планировщик потоков выделяет процессор для потока в зависимости от его приоритета. Когда бы мы ни создавали поток в Java, он всегда имеет некоторый приоритет. Приоритет может быть задан либо JVM при создании потока, либо программистом в явном виде.

Приоритеты в потоках - это концепция, в которой каждый поток имеет свой приоритет, что на языке обывателей означает, что каждый объект имеет здесь свой приоритет, который представлен числами от 1 до 10.

- По умолчанию приоритет установлен равным 5, как и положено.
- Минимальный приоритет равен 1.
- Максимальный приоритет равен 10.

Здесь определены 3 константы, а именно:

- `public static int NORM_PRIORITY`
- `public static int MIN_PRIORITY`
- `public static int MAX_PRIORITY`

Как получить и установить приоритет потока в Java:

- **public final int getPriority()** - метод возвращает приоритет заданного потока.
- **public final void setPriority(int newPriority)** - метод изменяет приоритет потока на значение `newPriority`. Этот метод выбрасывает `IllegalArgumentException`, если значение параметра `newPriority` выходит за пределы минимального (1 - `MIN_PRIORITY`) и максимального (10 - `MAX_PRIORITY`) ограничений.

Переведено с помощью www.DeepL.com/Translator (бесплатная версия)

Источник: <https://www.geeksforgeeks.org/java-thread-priority-multithreading/>

Основное задание:

Реализуйте систему многопоточности, в которой присутствует как минимум 3 потока и 3 группы потока, в которых как минимум одна является подгруппой одной из двух.

По завершению работы, составьте отчет, в котором должно быть – Ваша фамилия, имя, группа, тема работы, Ваш вариант для реализации задания, краткое описание реализации задания, ссылку на исходный код на GitHub. Исходный код push-ите в вашу ветку в соответствующем репозитории - <https://github.com/FCIM-PCD/Practice-Work-RU>. Сохранить отчет в формате PDF и отправить на ELSE - <https://else.fcim.utm.md/mod/assign/view.php?id=5826>.