MD5.1 Жизненный цикл сервиса

Цель работы

Познакомиться с жизненным циклом связанного и свободного сервиса в ОС Android, увидеть на практике порядок вызова методов жизненного цикла.

Задания для выполнения

1. Создайте в приложении сервис, переопределив все основные методы жизненного цикла. Каждый метод должен выводить в консоль свое название при вызове.
2. Не забудьте вызов методов суперкласса и объявление сервиса в манифесте приложения.
3. В основной активности создайте две конопки. При нажатии на первую интент, вызывающий созданный сервис (явный интент) рассылается методом startService(), при нажатии второй - методом bindService().
4. Запустите приложение и понаблюдайте за жизненным циклом сервиса.

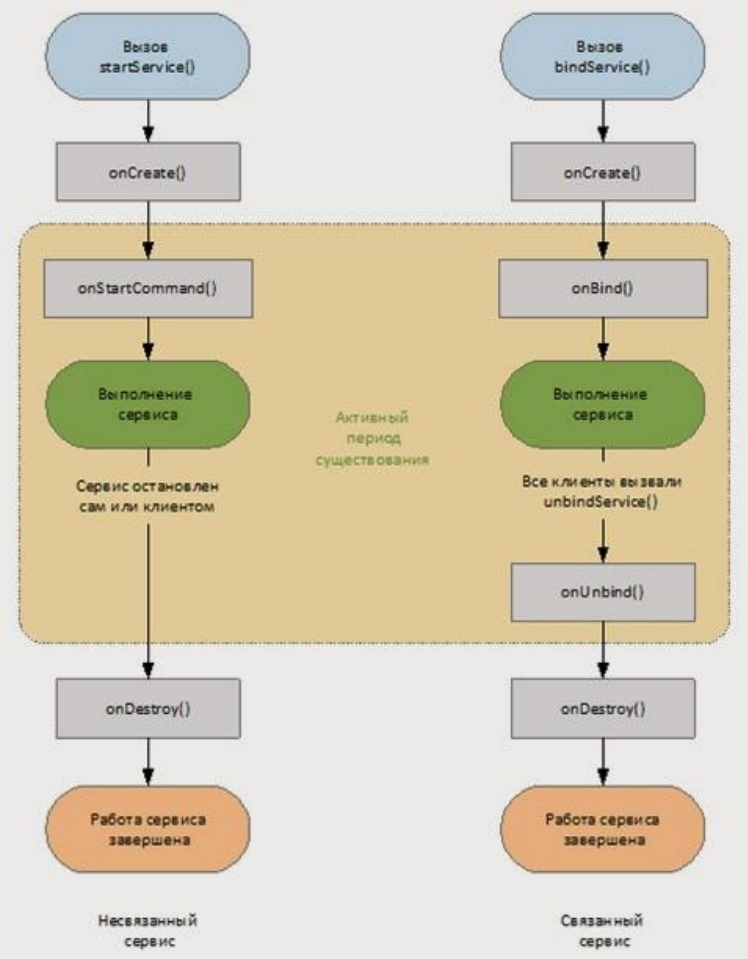
Контрольные вопросы

5. Когда происходит создание и удаление связанного сервиса?

Ответ: Создание связанного происходит при вызове метода bindService (intent, connection, context), удаление при вызове unbindService(connection)

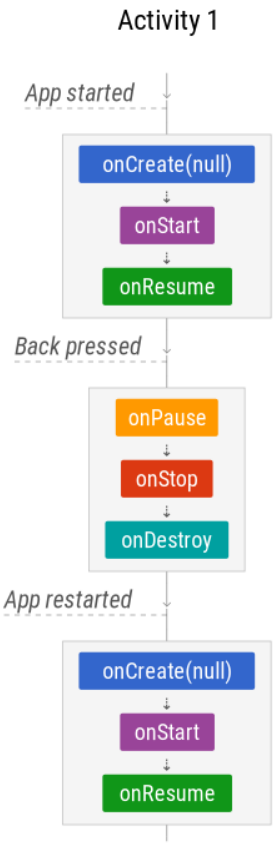
6. Какие методы жизненного цикла вызываются у сервиса и в какой последовательности?

Ответ: При создании связанного сервиса - onCreate(), onBind(). При использовании startService() - onStartCommand()



7. Как соотносится ЖЦ активности и сервиса?

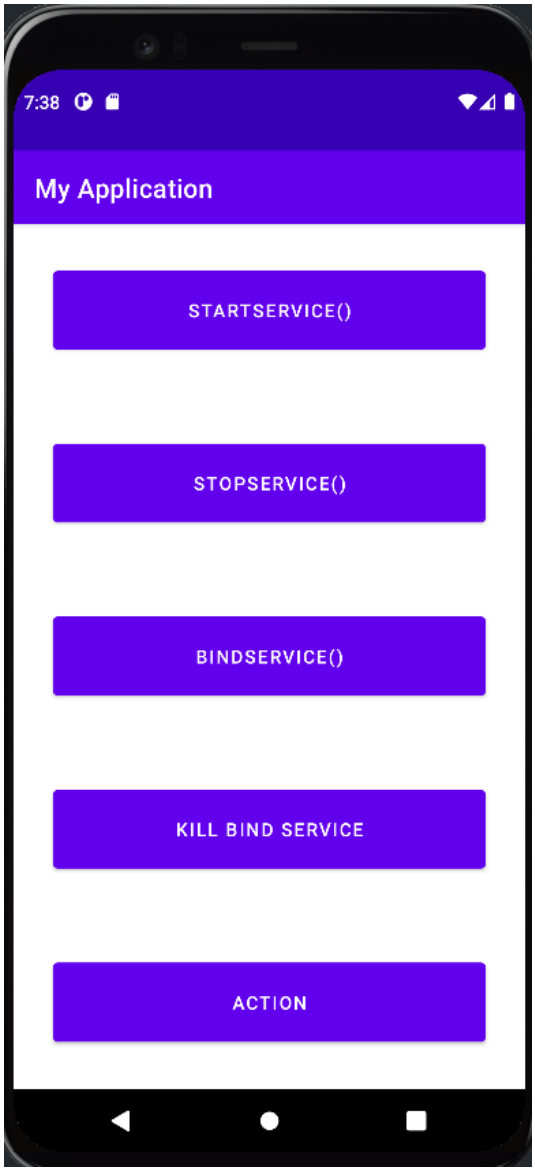
Ответ: Есть методы onCreate(), onDestroy()



Дополнительные задания

8. Добавьте в сервис полезное фоновое действие, которое выполняется продолжительное время. Как изменилось поведение сервиса?
9. Добавьте возможность из клиента (активности) убить сервис.
10. Добавьте логирование методов жизненного цикла активности. Пронаблюдайте за ней.

Работа приложения:



При нажатии на кнопку startService() сервис запускается не связанно и происходит вызов метода ЖЦ сервиса onStartCommand

```
2022-02-16 19:39:22.800 3930-3930/com.example.myapplication D/SERVICE: onCreate() Service has started
2022-02-16 19:39:22.801 3930-3930/com.example.myapplication D/SERVICE: onStartCommand: Intent { cmp=com.example.myapplication/.MyService }0 startId: 1
```

При нажатии на stopService() сервис завершает свою работу и происходит вызов метода onDestroy()

```
2022-02-16 19:39:49.145 3930-3930/com.example.myapplication D/SERVICE: onDestroy() Service has been destroyed!
```

При нажатии на кнопку bindService() происходит создание связанного сервиса и подключение

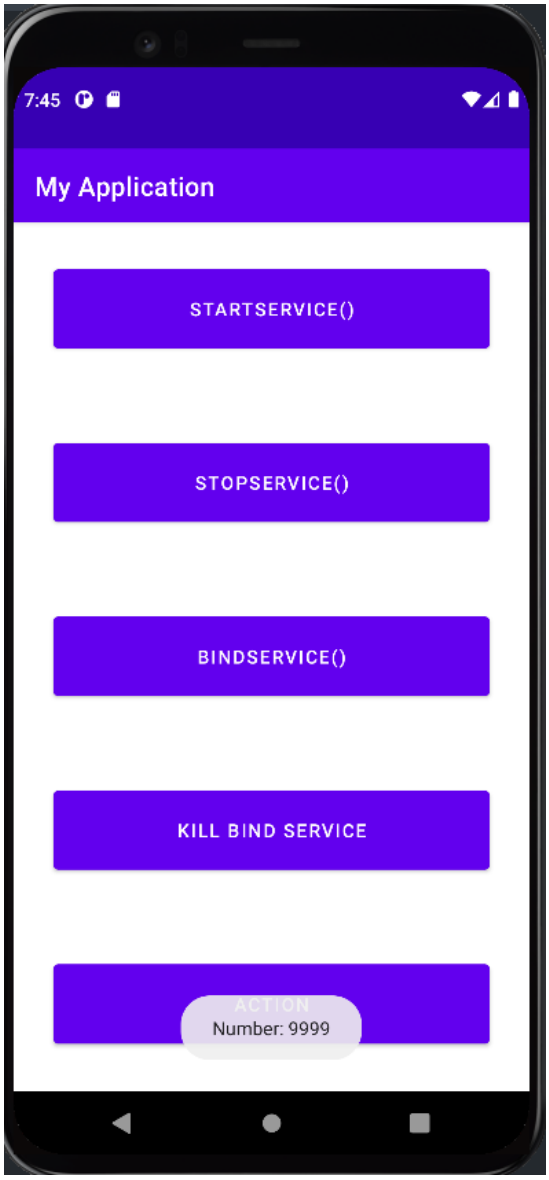
```
2022-02-16 19:40:24.416 3930-3930/com.example.myapplication D/SERVICE: onBind() Service binded
```

Кнопка ACTION выполняет полезное действие, генерирует рандомные числа от 0 до 10000 пока число не станет больше или равно 9999

```
2022-02-16 19:41:14.715 3930-3930/com.example.myapplication D/SERVICE: genRandomNumber(): 393
2022-02-16 19:41:14.715 3930-3930/com.example.myapplication D/SERVICE: genRandomNumber(): 533
2022-02-16 19:41:14.715 3930-3930/com.example.myapplication D/SERVICE: genRandomNumber(): 9742
2022-02-16 19:41:14.715 3930-3930/com.example.myapplication D/SERVICE: genRandomNumber(): 1931
2022-02-16 19:41:14.715 3930-3930/com.example.myapplication D/SERVICE: genRandomNumber(): 2882
2022-02-16 19:41:14.715 3930-3930/com.example.myapplication D/SERVICE: genRandomNumber(): 3100
2022-02-16 19:41:14.716 3930-3930/com.example.myapplication D/SERVICE: genRandomNumber(): 4273
2022-02-16 19:41:14.716 3930-3930/com.example.myapplication D/SERVICE: genRandomNumber(): 9440
2022-02-16 19:41:14.716 3930-3930/com.example.myapplication D/SERVICE: genRandomNumber(): 3270
2022-02-16 19:41:14.716 3930-3930/com.example.myapplication D/SERVICE: genRandomNumber(): 216
2022-02-16 19:41:14.716 3930-3930/com.example.myapplication D/SERVICE: genRandomNumber(): 6188
2022-02-16 19:41:14.716 3930-3930/com.example.myapplication D/SERVICE: genRandomNumber(): 3872
2022-02-16 19:41:14.716 3930-3930/com.example.myapplication D/SERVICE: genRandomNumber(): 9721
2022-02-16 19:41:14.716 3930-3930/com.example.myapplication D/SERVICE: genRandomNumber(): 386
2022-02-16 19:41:14.716 3930-3930/com.example.myapplication D/SERVICE: genRandomNumber(): 9826
2022-02-16 19:41:14.716 3930-3930/com.example.myapplication D/SERVICE: genRandomNumber(): 411
2022-02-16 19:41:14.716 3930-3930/com.example.myapplication D/SERVICE: genRandomNumber(): 3395
2022-02-16 19:41:14.716 3930-3930/com.example.myapplication D/SERVICE: genRandomNumber(): 995
2022-02-16 19:41:14.716 3930-3930/com.example.myapplication D/SERVICE: genRandomNumber(): 9790
2022-02-16 19:41:14.716 3930-3930/com.example.myapplication D/SERVICE: genRandomNumber(): 3662
2022-02-16 19:41:14.716 3930-3930/com.example.myapplication D/SERVICE: genRandomNumber(): 9999
```

```
2022-02-16 19:41:14.716 3930-3930/com.example.myapplication D/SERVICE: genRandomNumber(): 9826
2022-02-16 19:41:14.716 3930-3930/com.example.myapplication D/SERVICE: genRandomNumber(): 411
2022-02-16 19:41:14.716 3930-3930/com.example.myapplication D/SERVICE: genRandomNumber(): 3395
2022-02-16 19:41:14.716 3930-3930/com.example.myapplication D/SERVICE: genRandomNumber(): 995
2022-02-16 19:41:14.716 3930-3930/com.example.myapplication D/SERVICE: genRandomNumber(): 9790
2022-02-16 19:41:14.716 3930-3930/com.example.myapplication D/SERVICE: genRandomNumber(): 3662
2022-02-16 19:41:14.716 3930-3930/com.example.myapplication D/SERVICE: genRandomNumber(): 9999
2022-02-16 19:41:42.825 3930-3930/com.example.myapplication D/SERVICE: MyService onUnbind
2022-02-16 19:41:42.826 3930-3930/com.example.myapplication D/SERVICE: onDestroy() Service has been destroyed!
```

Также на главном экране высвечивается дополнительная информация о работе функции через Toast


7:45
My Application
STARTSERVICE()
STOPSERVICE()
BINDSERVICE()
KILL BIND SERVICE
ACTION
Number: 9999