**Лабораторная работа №2**

## Программное обновление элементов отображения

**Цель работы:** Познакомиться с механизмами доступа к элементам загруженного отображения и изменения их свойств.

**Задание:** Модернизировать существующую лабораторную работу №1, добавить к отображаемой информации количество дней до следующего дня рождения автора, правильно обработать ситуацию, когда день рождения – сегодня.

Ход работы:

1. Получаем дату дня рождения в этом году и сегодняшнюю

SimpleDateFormat formatter = new SimpleDateFormat("dd.MM.yyyy");  
  
int year = Year.*now*().getValue();  
LocalDate birthday = LocalDate.*of*(year, Month.***DECEMBER***, 20);  
LocalDate today = LocalDate.*now*();

1. Обрабатываем случай когда день рождение сегодня(разница в 0 дней)

long d = Duration.*between*(today.atStartOfDay(), birthday.atStartOfDay()).toDays();  
if(d==0){  
 tv.setText("HAPPY BIRTHDAY!");  
 return;  
}  
if(d<0){  
 birthday = LocalDate.*of*(year+1, Month.***DECEMBER***, 20);  
}

1. Переводим дату дня рождения в миллисекунды, а также получаем текущую дату в миллисекундах.

String endTime = birthday.format(DateTimeFormatter.*ofPattern*("dd.MM.yyyy"));  
long milliseconds=0;  
  
final CountDownTimer mCountDownTimer;  
  
Date endDate;  
try {  
 endDate = formatter.parse(endTime);  
 milliseconds = endDate.getTime();  
} catch (ParseException e) {  
 e.printStackTrace();  
}  
  
final long[] startTime = {System.*currentTimeMillis*()};

1. Создаём таймер отсчёта который будет тикать раз 1000мс (1 секунду)

mCountDownTimer = new CountDownTimer(milliseconds, 1000) {  
 @Override  
 public void onTick(long millisUntilFinished) {  
  
 startTime[0] = startTime[0] -1;  
 Long serverUptimeSeconds =  
 (millisUntilFinished - startTime[0]) / 1000;  
  
 String res = String.*format*("%d", serverUptimeSeconds / 86400)+"D ";  
 res += String.*format*("%d", (serverUptimeSeconds % 86400) / 3600)+"H ";  
 res += String.*format*("%d", ((serverUptimeSeconds % 86400) % 3600) / 60)+"M ";  
 res += String.*format*("%d", ((serverUptimeSeconds % 86400) % 3600) % 60)+"S";  
 tv.setText(res);  
  
 }  
  
 @Override  
 public void onFinish() {  
  
 }  
}.start();

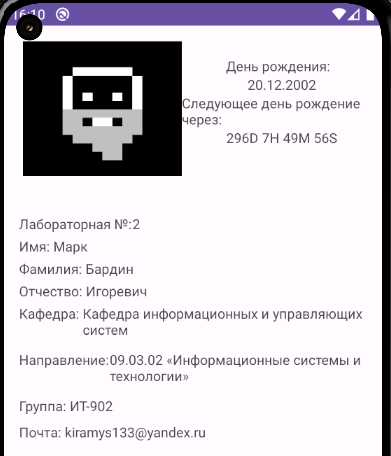


Рисунок 1 - результат работы

**Вывод:** познакомились с с механизмами доступа к элементам загруженного отображения и изменения их свойств.