```
private boolean isDeviceSupportCamera() {
    if (getApplicationContext().getPackageManager().hasSystemFeature(
        PackageManager.FEATURE_CAMERA)) {
      // this device has a camera
      return true;
    } else {
      // no camera on this device
      return false;
    }
  }
  private void recordVideo() {
    Intent intent = new Intent(MediaStore.ACTION_VIDEO_CAPTURE);
    fileUri = getOutputMediaFileUri(MEDIA_TYPE_VIDEO);
    // set video quality
    intent.putExtra(MediaStore.EXTRA_VIDEO_QUALITY, 1);
    intent.putExtra(MediaStore.EXTRA_OUTPUT, fileUri);
    startActivityForResult(intent, CAMERA_CAPTURE_VIDEO_REQUEST_CODE);
  }
  @Override
  protected void onActivityResult(int requestCode, int resultCode, Intent data) {
    if (requestCode == CAMERA_CAPTURE_IMAGE_REQUEST_CODE) {
      if (resultCode == RESULT_OK) {
      } else if (resultCode == RESULT_CANCELED) {
        Toast.makeText(getApplicationContext(),
            "User cancelled image capture", Toast.LENGTH_SHORT)
            .show();
      } else {
        Toast.makeText(getApplicationContext(),
            "Sorry! Failed to capture image", Toast.LENGTH_SHORT)
            .show();
      }
    } else
      if (requestCode == CAMERA_CAPTURE_VIDEO_REQUEST_CODE) {
      if (resultCode == RESULT_OK) {
      } else if (resultCode == RESULT_CANCELED) {
        Toast.makeText(getApplicationContext(),
            "User cancelled video recording", Toast.LENGTH_SHORT)
            .show();
      } else {
        Toast.makeText(getApplicationContext(),
            "Sorry! Failed to record video", Toast.LENGTH_SHORT)
            .show();
      }
    }
  }
  public Uri getOutputMediaFileUri(int type) {
```