Прием картинок из интернета

есть картинка на html странице, как мне загрузить её в приложение и показать в ImageView

Хороший пример с наследованием загрузчика картинки от [AsyncTask](http://developer.android.com/reference/android/os/AsyncTask.html) и определением своего метода [doInBackground()](http://developer.android.com/reference/android/os/AsyncTask.html" \l "doInBackground(Params...)), взятый с [английского SO](http://stackoverflow.com/questions/2471935/how-to-load-an-imageview-by-url-in-android), который взял его с Android Developers, который уже его убрал.

Сам AsyncTask представляет собой реализацию короткого асинхронного запроса для основного GUI потока без необходимости самому ковырятся с потоками.

// Показать картинку

new DownloadImageTask((ImageView) findViewById(R.id.imageView1))

.execute("http://java.sogeti.nl/JavaBlog/wp-content/uploads/2009/04/android\_icon\_256.png");

}

public void onClick(View v) {

startActivity(new Intent(this, IndexActivity.class));

finish();

}

private class DownloadImageTask extends AsyncTask<String, Void, Bitmap> {

ImageView bmImage;

public DownloadImageTask(ImageView bmImage) {

this.bmImage = bmImage;

}

protected Bitmap doInBackground(String... urls) {

String urldisplay = urls[0];

Bitmap mIcon11 = null;

try {

InputStream in = new java.net.URL(urldisplay).openStream();

mIcon11 = BitmapFactory.decodeStream(in);

} catch (Exception e) {

Log.e("Ошибка передачи изображения", e.getMessage());

e.printStackTrace();

}

return mIcon11;

}

protected void onPostExecute(Bitmap result) {

bmImage.setImageBitmap(result);

}

}

И в манифесте для приложения нужно конечно же разрешить загрузку:

<uses-permission android:name="android.permission.INTERNET" />

Если нужно быстро и без всякой асинхронности, обработки ошибок и.т.д.:

URL newurl = new URL(image\_location\_url);

mIcon\_val = BitmapFactory.decodeStream(newurl.openConnection() .getInputStream());

myImageView.setImageBitmap(mIcon\_val);

Для совсем ленивых однострочник на [Picasso](http://square.github.io/picasso/):

Picasso.with(context).load(myImageURL).into(imageView);