20\_07\_20 학습 내용

RTSP Player 로그인 구현 중

처음에는 Login.class에 메소드 하나에다가 Http Get요청을 때려 박았는데

android.os.NetworkOnMainThreadException 과 같은 에러가 발생하였다.

찾아본 결과 메인 스레드에서 동작 시켜서 발생하는 에러로,

추가로 Thread를 생성하여 그걸로 돌리는 해결 되었다.

public class Login extends Activity

{

TextView ResultHtml;

@Override

protected void onCreate(Bundle savedInstanceState) {

super.onCreate(savedInstanceState);

setContentView(R.layout.*activity\_login*);

// Thread로 웹서버에 접속

new Thread() {

public void run() {

String naverHtml = getNaverHtml();

Bundle bun = new Bundle();

bun.putString("Test", naverHtml);

Message msg = handler.obtainMessage();

msg.setData(bun);

handler.sendMessage(msg);

}

}.start();

ResultHtml = (TextView)this.findViewById(R.id.*result*);

//tvNaverHtml.setText(naverHtml);

}

private String getNaverHtml(){

String naverHtml = "";

URL url =null;

HttpURLConnection http = null;

InputStreamReader isr = null;

BufferedReader br = null;

try{

url = new URL(" "); //URL은 배포 주소 넣으시면 됩니다.

http = (HttpURLConnection) url.openConnection();

http.setConnectTimeout(3\*1000);

http.setReadTimeout(3\*1000);

isr = new InputStreamReader(http.getInputStream());

br = new BufferedReader(isr);

String str = null;

while ((str = br.readLine()) != null) {

naverHtml += str + "\n";

}

}catch(Exception e){

}finally{

if(http != null){

try{http.disconnect();}catch(Exception e){}

}

if(isr != null){

try{isr.close();}catch(Exception e){}

}

if(br != null){

try{br.close();}catch(Exception e){}

}

}

return naverHtml;

}

Handler handler = new Handler() {

public void handleMessage(Message msg) {

Bundle bun = msg.getData();

String naverHtml = bun.getString("Test");

ResultHtml.setText(naverHtml);

}

};

}

결과 화면은 아래와 같다.

→ URL로 접속 했을때의 결과가 똑같이 나옴을 알 수 있다.

|  |  |
| --- | --- |
|  |  |

이를 토대로 내일 파라미터에 ID와 Passwd를 넣어서 결과를 반환 할 수 있도록

API Gateway를 손보면 될 것 같다.