

class Task1 extends AsyncTask<String, Void, String> {

```
String sendMsg, receiveMsg;
String serverip = ServerInfo.SERVER_IP + "nickname_check";

@Override
protected String doInBackground(String... strings) {
    try {
        String str = "";
        URL url = new URL(serverip);

        //서버연결
        HttpURLConnection conn = (HttpURLConnection) url.openConnection();
        conn.setRequestMethod("POST");

        OutputStreamWriter osw = new OutputStreamWriter(conn.getOutputStream());

        //서버로 전달할 내용
        sendMsg = strings[0];

        //서버로 값 전송
        osw.write(sendMsg);
        osw.flush();

        //서버로 값 전송이 완료되면 서버에서 처리한 결과를 받는다
        //getResponseCode() : 200 -> 정상
        //getResponseCode() : 404,500 -> 비정상
        if (conn.getResponseCode() == conn.HTTP_OK) {
            InputStreamReader tmp = new InputStreamReader(conn.getInputStream(), "utf-8");
            BufferedReader reader = new BufferedReader(tmp);
            StringBuffer buffer = new StringBuffer();

            while ((str = reader.readLine()) != null) {
                buffer.append(str);
            }

            //서버에서 넘겨준 JSON 형식의 결과값
            receiveMsg = buffer.toString();

            JSONArray jarray = new JSONObject(receiveMsg).getJSONArray("res");
            JSONObject jobject = jarray.getJSONObject(0);
            String result = jobject.optString("result");

            if (result.equals("can")) {
                receiveMsg = "yes";
            } else {
                receiveMsg = "no";
            }
            Log.i("돌아온 값", receiveMsg);
        }

    } catch (Exception e) {

    }

    return receiveMsg;
}
```

https://localhost:8080/senghan1992/nickname_check

serverip를 URL로 인코딩하고
HttpURLConnection을 이용해서 연결을 연다
sendMsg는 서버로 보낼 Prameter를 의미한다
osw.write(sendMsg)를 통해 서버로 보내고
osw.flush()로 새로고침(?) 같은 느낌으로 보내준다

값이 제대로 돌아온다면 getResponseCode()가 200일 것이고 나머지는
문제가 있는것이니....Googleing열심히 하시길...
InputStreamReader로 서버에서 보내준 것들을 읽어들인다
읽어들인것을 String화 해서 넣어준 것이 receiveMsg이다
receiveMsg를 Json형식으로 바꿔서 사용해준다

```
@Override
protected void onPostExecute(String s) {
    //Toast.makeText(getApplicationContext(), s, Toast.LENGTH_SHORT).show();
    if (s.equals("yes")){
        is_checked_nickname = true;
        btn_join.setEnabled(true);
        edit_nickname.setCompoundDrawablesWithIntrinsicBounds(R.mipmap.join_slash,0,
            R.mipmap.after_check,0);
    }else{
        is_checked_nickname = false;
        edit_nickname.setCompoundDrawablesWithIntrinsicBounds(R.mipmap.join_slash,0,
            R.mipmap.after_check2,0);
    }
}
```

doInBackground에서 receiveMsg를 return 해주면
onPostExecute(String s)중에 s로 그 값이 들어오게 되고
그것을 활용해서 작업 후 또다른 작업을 정의해 주면 된다

```
@RequestMapping("/mobile/nickname_check")
@ResponseBody
public String join_check(HttpServletRequest request) {
    Map<String, Object> map = new HashMap<String, Object>();
    map.put("request", request);

    MemberVO vo = service.check(map);
    String resultStr = "";
    if (vo == null) {
        resultStr = String.format("{res:[{result:'%s'}]}", "can");
    } else {
        resultStr = String.format("{res:[{result:'%s'}]}", "cant");
    }
    return resultStr;
}
```

여기로 들어가있음

이것이 굉장히 중요!!!
이렇게 해줘야 JSON 타입으로 받을 수가 있다