```xml
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
xmlns:app="http://schemas.android.com/apk/res-auto"
xmlns:tools="http://schemas.android.com/tools"
android:layout_width="match_parent"
android:layout_height="match_parent"
android:orientation="vertical"
tools:context=".MainActivity11">

<LinearLayout
android:layout_width="match_parent"
android:layout_height="wrap_content"
android:orientation="horizontal">

<Button
android:id="@+id/web_btn1"
android:layout_width="wrap_content"
android:layout_height="wrap_content"
android:layout_weight="1"
android:text="xml연습" />

<Button
android:id="@+id/web_btn2"
android:layout_width="wrap_content"
android:layout_height="wrap_content"
android:layout_weight="1"
android:text="json연습" />

</LinearLayout>

<ScrollView
android:layout_width="match_parent"
android:layout_height="match_parent">
<WebView
android:id="@+id/webView3"
android:layout_width="match_parent"
android:layout_height="match_parent" />

</ScrollView>
</LinearLayout>
```

XML연습	JSON연습
WebView	

```java
public class MainActivity11 extends AppCompatActivity {
Button web_btn1, web_btn2 ;
WebView webView3 ;

Handler handler = new Handler();
String year, month, day, hour;
final String WEATHER_URL = "http://www.kma.go.kr/XML/weather/sfc_web_map.xml";
ArrayList<VO> list = new ArrayList<>();

String json_mag ;
@Override
protected void onCreate(Bundle savedInstanceState) {
super.onCreate(savedInstanceState);
setContentView(R.layout.activity_main11);

web_btn1 = findViewById(R.id.web_btn1);
web_btn2 = findViewById(R.id.web_btn2);
webView3 = findViewById(R.id.webView3);

web_btn1.setOnClickListener(new View.OnClickListener() {
@Override
public void onClick(View v) {
new Thread(new Runnable() {
@Override
public void run() {
serverConnect(getStreamFromUrl());
handler.post(new Runnable() {
@Override
public void run() {
xml_process();
}
});
}
}).start();
}
});

web_btn2.setOnClickListener(new View.OnClickListener() {
@Override
public void onClick(View v) {

new Thread(new Runnable() {
@Override
public void run() {
serverConnect2("http://openapi.seoul.go.kr:8088/sample/json/SeoulLibraryTime/1/5/");

handler.post(new Runnable() {
@Override
public void run() {
json_process();
}
});
}
}).start();
}
});
}

// 해당 사이트에 접속하는 InputStgream 리턴
private InputStream getStreamFromUrl(){
InputStream in = null;
String msg = null;
HttpURLConnection connection = null;
try {
URL url = new URL(WEATHER_URL);
connection = (HttpURLConnection)url.openConnection();
in = connection.getInputStream();
}catch (Exception e){
}
return  in;
}
// XmlPullParser 에 InputStgream을 넣어서 사용
private void serverConnect(InputStream stream){
String local ="", desc="", ta="", icon ="";
try {
XmlPullParserFactory factory = XmlPullParserFactory.newInstance();
XmlPullParser parser = factory.newPullParser();
parser.setInput(stream, "UTF-8");

// 문서가 끝날때 까지 반복 수행하라
while (parser.getEventType() != XmlPullParser.END_DOCUMENT){

if(parser.getEventType() == XmlPullParser.START_TAG){    // 시작 태그
if(parser.getName().equals("weather")){
year = parser.getAttributeValue(0);
month = parser.getAttributeValue(1);
day = parser.getAttributeValue(2);
hour = parser.getAttributeValue(3);
}
if(parser.getName().equals("local")){
icon = parser.getAttributeValue(1);
desc = parser.getAttributeValue(2);
ta = parser.getAttributeValue(3);
}
}else if(parser.getEventType() == XmlPullParser.TEXT){    // 텍스트
local = parser.getText();
}else if(parser.getEventType() == XmlPullParser.END_TAG){ // 끝 태그
if(! local.trim().equals("")){
list.add(new VO(local, desc, ta, icon));
}
}
parser.next();

}catch (Exception e){
}
}
```

```java
private void xml_process(){
StringBuffer sb = new StringBuffer();
sb.append("<html><body><h2 align='center'> ** 기상청 제공 날씨 정보 ** </h2>");
sb.append( year+ "년 " +month+"월 "+day+"일 "+ hour+"시 현재 <br>");
sb.append("<table width='100%' border='1'>");
sb.append("<thead>");
sb.append("<tr><th>지역</th><th>상태</th><th>온도</th><th>이미지</th></tr>");
sb.append("</thead>");
sb.append("<tbody align='center'>");
for (VO k : list){
sb.append("<tr>");
sb.append("<td>"+k.getLocal()+"</td>");
sb.append("<td>"+k.getDesc()+"</td>");
sb.append("<td>"+k.getTa()+"</td>");
if(k.getIcon().equals("")){
sb.append("<td> - </td>");
}else{
sb.append("<td><img
src='http://www.kma.go.kr/images/icon/NW/NB"+k.getIcon()+".png'></td>");
}
sb.append("</tr>");
}
sb.append("</tbody></table></body></html>");

webView3.loadData(sb.toString(),"text/html","utf-8");
}

private void serverConnect2(String str){
StringBuffer sb = new StringBuffer();
try {
URL url = new URL(str);
HttpURLConnection conn = (HttpURLConnection)url.openConnection();
if(conn !=null){
conn.setConnectTimeout(10000);
if(conn.getResponseCode() == HttpURLConnection.HTTP_OK){
BufferedReader br =
new BufferedReader(new InputStreamReader(conn.getInputStream(),"utf-8"));
while(true){
String line = br.readLine();
if(line == null) break;
sb.append(line+"\n");
}
br.close();
}
conn.disconnect();
}
}catch (Exception e){
}
json_mag = sb.toString();
}

private void json_process(){
try {
StringBuffer sb = new StringBuffer();
sb.append("<html><body><h2 align='center'> ** 서울지역 도서관 정보 ** </h2>");
sb.append("<table width='100%' border='1'>");
sb.append("<thead>");
sb.append("<tr><th>도서관이름</th><th>주소</th><th>휴관일</th><th>전화번호</th></tr>");
sb.append("</thead>");
sb.append("<tbody align='center'>");
BufferedReader br =
new BufferedReader(new StringReader(json_mag));
String res = br.readLine();

JSONObject j_obj1 = new JSONObject(res);
JSONObject j_obj2 = new JSONObject(j_obj1.getString("SeoulLibraryTime"));
JSONArray j_arr = new JSONArray(j_obj2.getString("row"));
for (int i=0; i <j_arr.length(); i++){
sb.append("<tr>");
sb.append("<td>" + j_arr.getJSONObject(i).getString("LBRRY_NAME") +"</td>");
sb.append("<td>"+ j_arr.getJSONObject(i).getString("ADRES") +"</td>");
sb.append("<td>"+ j_arr.getJSONObject(i).getString("FDRM_CLOSE_DATE") +"</td>");
sb.append("<td>"+ j_arr.getJSONObject(i).getString("TEL_NO") +"</td>");
sb.append("</tr>");
}
sb.append("</tbody></table></body></html>");
webView3.loadData(sb.toString(),"text/html","utf-8");
}catch (Exception e){
}
}
}
```