3.1 Crud

addEmp.php

|  |
| --- |
| <?php  if($\_SERVER['REQUEST\_METHOD']=='POST'){  $name = $\_POST['name'];  $desg = $\_POST['desg'];  $sal =$\_POST['salary'];  $sql="INERT INTO emplotee(name,designation,salary) VALUES('$name','$desg','$sal')";  require\_once('dbConnext.php');  if(mysqli\_query($con,$sql)){  echo 'Employee Added Successfully';  } else {  echo 'Could Not Add Employee';  }  mysqli\_close($con);  }  ?> |

getAllEmp.php

|  |
| --- |
| <?php  require\_once('dbConnect.php');  $sql ="SELECT \* FROM employee";  $r = mysqli\_query($con,$sql);  $result = array();  while ($row = mysqli\_fetch\_arrray($r)) {  array\_push ($result,array("id"=>$row['id'],"name=>$row['name']));  }  echo json\_encode(array('result'=>$result));  mysqli\_close($con);  ?> |

dbConnect.php

|  |
| --- |
| <?php  define('HOST','localhost');  define('USER','root');  define('PASS','');  define('DB','crud');  $con=mysqli\_connect(HOST,USER,PASS,DB) or die('Unable to Connect');  ?> |

getEmp.php

|  |
| --- |
| <?php  $id =$\_GET['id'];  require\_once('dbConnect.php');  $sql="SELECT \* FROM employee WHERE id=$id";  $r = mysqli\_query($con,$sql);  $result = array();  $row = mysqli\_fetch\_arrray($r);  array\_push($result, array("id"=>$row['id'],"name"=>$row['name'],"desg"=>$row['designation'],"salary"=>$row['salary']));  echo json\_encode(array('result'=>$result));  mysqli\_close($con);  ?> |

updateEmp.php

|  |
| --- |
| <?php  if($\_SERVER['REQUEST\_METHOD']=='POST'){  $id= $\_POST['id']  $name = $\_POST['name'];  $desg = $\_POST['desg'];  $sal =$\_POST['salary'];  require\_once('dbConnect.php');  $sql ="UPDATE employee SET name ='$name',designation='$desg', salary ='$sal' WHERE id =$id";  if (mysqli\_query($con,$sql)) {  echo 'Employee Update Successfully';  }else {  echo 'Could Not Update Employee Try Again';  }  mysqli\_close($con);  }  ?> |

deleteEmp.php

|  |
| --- |
| <?php  $id = $\_GET['id'];  require\_once('dbConnect.php');  $sql = "DELETE FROM employee WHERE id = $id";  if (mysqli\_query($con, $sql)) {  echo 'Employee Delete Successfully';  }else {  echo 'Could Not Delete Employee Try Again';  }  mysqli\_close($con);  ?> |

Config.java

|  |
| --- |
| package net.simplifiedcoding.mysqlcrud;   public class Config {    public static final String *URL\_ADD*="http://172.20.10.2/androcrud/addEmp.php";  public static final String *URL\_GET\_ALL* = "http://172.20.10.2/androcrud/getAllEmp.php";  public static final String *URL\_GET\_EMP* = "http://172.20.10.2/androcrud/getEmp.php?id=";  public static final String *URL\_UPDATE\_EMP* = "http://172.20.10.2/androcrud/updateEmp.php";  public static final String *URL\_DELETE\_EMP* = "http://172.20.10.2/androcrud/deleteEmp.php?id=";  public static final String *URL\_SELECT\_EMP* = "http://172.20.10.2/androcrud/selectEmp.php?id=";      public static final String *KEY\_EMP\_ID* = "id";  public static final String *KEY\_EMP\_NAME* = "name";  public static final String *KEY\_EMP\_DESG* = "desg";  public static final String *KEY\_EMP\_SAL* = "salary";  public static final String *KEY\_EMP\_USERNAME* = "username";  public static final String *KEY\_EMP\_PASSWORD* = "password";     public static final String *TAG\_JSON\_ARRAY*="result";  public static final String *TAG\_ID* = "id";  public static final String *TAG\_NAME* = "name";  public static final String *TAG\_DESG* = "desg";  public static final String *TAG\_SAL* = "salary";       public static final String *EMP\_ID* = "emp\_id";  }  } |

RequestHandler.java

|  |
| --- |
| package net.simplifiedcoding.mysqlcrud;  import java.io.BufferedReader; import java.io.BufferedWriter; import java.io.InputStreamReader; import java.io.OutputStream; import java.io.OutputStreamWriter; import java.io.UnsupportedEncodingException; import java.net.HttpURLConnection; import java.net.URL; import java.net.URLEncoder; import java.util.HashMap; import java.util.Map;  import javax.net.ssl.HttpsURLConnection;   public class RequestHandler {    public String sendPostRequest(String requestURL,  HashMap<String, String> postDataParams) {  //Creating a URL  URL url;   StringBuilder sb = new StringBuilder();  try {   url = new URL(requestURL);   HttpURLConnection conn = (HttpURLConnection) url.openConnection();    conn.setReadTimeout(15000);  conn.setConnectTimeout(15000);  conn.setRequestMethod("POST");  conn.setDoInput(true);  conn.setDoOutput(true);   OutputStream os = conn.getOutputStream();    BufferedWriter writer = new BufferedWriter(  new OutputStreamWriter(os, "UTF-8"));  writer.write(getPostDataString(postDataParams));   writer.flush();  writer.close();  os.close();  int responseCode = conn.getResponseCode();   if (responseCode == HttpsURLConnection.*HTTP\_OK*) {   BufferedReader br = new BufferedReader(new InputStreamReader(conn.getInputStream()));  sb = new StringBuilder();  String response;   while ((response = br.readLine()) != null){  sb.append(response);  }  }   } catch (Exception e) {  e.printStackTrace();  }  return sb.toString();  }   public String sendGetRequest(String requestURL){  StringBuilder sb =new StringBuilder();  try {  URL url = new URL(requestURL);  HttpURLConnection con = (HttpURLConnection) url.openConnection();  BufferedReader bufferedReader = new BufferedReader(new InputStreamReader(con.getInputStream()));   String s;  while((s=bufferedReader.readLine())!=null){  sb.append(s+"\n");  }  }catch(Exception e){  }  return sb.toString();  }   public String sendGetRequestParam(String requestURL, String id){  StringBuilder sb =new StringBuilder();  try {  URL url = new URL(requestURL+id);  HttpURLConnection con = (HttpURLConnection) url.openConnection();  BufferedReader bufferedReader = new BufferedReader(new InputStreamReader(con.getInputStream()));   String s;  while((s=bufferedReader.readLine())!=null){  sb.append(s+"\n");  }  }catch(Exception e){  }  return sb.toString();  }   private String getPostDataString(HashMap<String, String> params) throws UnsupportedEncodingException {  StringBuilder result = new StringBuilder();  boolean first = true;  for (Map.Entry<String, String> entry : params.entrySet()) {  if (first)  first = false;  else  result.append("&");   result.append(URLEncoder.*encode*(entry.getKey(), "UTF-8"));  result.append("=");  result.append(URLEncoder.*encode*(entry.getValue(), "UTF-8"));  }   return result.toString();  } } |

Activity\_main.java

|  |
| --- |
| package net.simplifiedcoding.mysqlcrud;  import android.app.ProgressDialog; import android.content.Intent; import android.os.AsyncTask; import android.support.v7.app.AppCompatActivity; import android.os.Bundle; import android.view.Menu; import android.view.MenuItem; import android.view.View; import android.widget.Button; import android.widget.EditText; import android.widget.Toast;  import java.util.HashMap;  public class MainActivity extends AppCompatActivity implements View.OnClickListener{    private EditText editusername;  private EditText editpassword;    private Button buttonlogin;    @Override  protected void onCreate(Bundle savedInstanceState) {  super.onCreate(savedInstanceState);  setContentView(R.layout.*activity\_main*);    editusername = (EditText) findViewById(R.id.*editusername*);  editpassword = (EditText) findViewById(R.id.*editpassword*);   buttonlogin = (Button) findViewById(R.id.*buttonlogin*);     buttonlogin.setOnClickListener(this);   }      @Override  public void onClick(View v) {   if(v == buttonlogin){  startActivity(new Intent(this,menu.class));  }  } } |

ViewAllEmployee.java

|  |
| --- |
| package net.simplifiedcoding.mysqlcrud;  import android.app.ProgressDialog; import android.content.Intent; import android.os.AsyncTask; import android.support.v7.app.AppCompatActivity; import android.os.Bundle; import android.view.Menu; import android.view.MenuItem; import android.view.View; import android.widget.AdapterView; import android.widget.ListAdapter; import android.widget.ListView; import android.widget.SimpleAdapter; import android.widget.Toast;  import org.json.JSONArray; import org.json.JSONException; import org.json.JSONObject;  import java.util.ArrayList; import java.util.HashMap;  public class ViewAllEmployee extends AppCompatActivity implements ListView.OnItemClickListener {   private ListView listView;   private String JSON\_STRING;   @Override  protected void onCreate(Bundle savedInstanceState) {  super.onCreate(savedInstanceState);  setContentView(R.layout.*activity\_view\_all\_employee*);  listView = (ListView) findViewById(R.id.*listView*);  listView.setOnItemClickListener(this);  getJSON();  }    private void showEmployee(){  JSONObject jsonObject = null;  ArrayList<HashMap<String,String>> list = new ArrayList<HashMap<String, String>>();  try {  jsonObject = new JSONObject(JSON\_STRING);  JSONArray result = jsonObject.getJSONArray(Config.*TAG\_JSON\_ARRAY*);   for(int i = 0; i<result.length(); i++){  JSONObject jo = result.getJSONObject(i);  String id = jo.getString(Config.*TAG\_ID*);  String name = jo.getString(Config.*TAG\_NAME*);   HashMap<String,String> employees = new HashMap<>();  employees.put(Config.*TAG\_ID*,id);  employees.put(Config.*TAG\_NAME*,name);  list.add(employees);  }   } catch (JSONException e) {  e.printStackTrace();  }   ListAdapter adapter = new SimpleAdapter(  ViewAllEmployee.this, list, R.layout.*list\_item*,  new String[]{Config.*TAG\_ID*,Config.*TAG\_NAME*},  new int[]{R.id.*id*, R.id.*name*});   listView.setAdapter(adapter);  }   private void getJSON(){  class GetJSON extends AsyncTask<Void,Void,String>{   ProgressDialog loading;  @Override  protected void onPreExecute() {  super.onPreExecute();  loading = ProgressDialog.*show*(ViewAllEmployee.this,"Fetching Data","Wait...",false,false);  }   @Override  protected void onPostExecute(String s) {  super.onPostExecute(s);  loading.dismiss();  JSON\_STRING = s;  showEmployee();  }   @Override  protected String doInBackground(Void... params) {  RequestHandler rh = new RequestHandler();  String s = rh.sendGetRequest(Config.*URL\_GET\_ALL*);  return s;  }  }  GetJSON gj = new GetJSON();  gj.execute();  }   @Override  public void onItemClick(AdapterView<?> parent, View view, int position, long id) {  Intent intent = new Intent(this, ViewEmployee.class);  HashMap<String,String> map =(HashMap)parent.getItemAtPosition(position);  String empId = map.get(Config.*TAG\_ID*).toString();  intent.putExtra(Config.*EMP\_ID*,empId);  startActivity(intent);  } } |

ViewEmployee.java

|  |
| --- |
| package net.simplifiedcoding.mysqlcrud;  import android.app.ProgressDialog; import android.content.DialogInterface; import android.content.Intent; import android.os.AsyncTask; import android.support.v7.app.AlertDialog; import android.support.v7.app.AppCompatActivity; import android.os.Bundle; import android.view.Menu; import android.view.MenuItem; import android.view.View; import android.widget.Button; import android.widget.EditText; import android.widget.Toast;  import org.json.JSONArray; import org.json.JSONException; import org.json.JSONObject;  import java.util.HashMap;  public class ViewEmployee extends AppCompatActivity implements View.OnClickListener {   private EditText editTextId;  private EditText editTextName;  private EditText editTextDesg;  private EditText editTextSalary;   private Button buttonUpdate;  private Button buttonDelete;    private String id;   @Override  protected void onCreate(Bundle savedInstanceState) {  super.onCreate(savedInstanceState);  setContentView(R.layout.*activity\_view\_employee*);   Intent intent = getIntent();   id = intent.getStringExtra(Config.*EMP\_ID*);   editTextId = (EditText) findViewById(R.id.*editTextId*);  editTextName = (EditText) findViewById(R.id.*editTextName*);  editTextDesg = (EditText) findViewById(R.id.*editTextDesg*);  editTextSalary = (EditText) findViewById(R.id.*editTextSalary*);   buttonUpdate = (Button) findViewById(R.id.*buttonUpdate*);  buttonDelete = (Button) findViewById(R.id.*buttonDelete*);    buttonUpdate.setOnClickListener(this);  buttonDelete.setOnClickListener(this);   editTextId.setText(id);   getEmployee();  }   private void getEmployee(){  class GetEmployee extends AsyncTask<Void,Void,String>{  ProgressDialog loading;  @Override  protected void onPreExecute() {  super.onPreExecute();  loading = ProgressDialog.*show*(ViewEmployee.this,"Fetching...","Wait...",false,false);  }   @Override  protected void onPostExecute(String s) {  super.onPostExecute(s);  loading.dismiss();  showEmployee(s);  }   @Override  protected String doInBackground(Void... params) {  RequestHandler rh = new RequestHandler();  String s = rh.sendGetRequestParam(Config.*URL\_GET\_EMP*,id);  return s;  }  }  GetEmployee ge = new GetEmployee();  ge.execute();  }   private void showEmployee(String json){  try {  JSONObject jsonObject = new JSONObject(json);  JSONArray result = jsonObject.getJSONArray(Config.*TAG\_JSON\_ARRAY*);  JSONObject c = result.getJSONObject(0);  String name = c.getString(Config.*TAG\_NAME*);  String desg = c.getString(Config.*TAG\_DESG*);  String sal = c.getString(Config.*TAG\_SAL*);   editTextName.setText(name);  editTextDesg.setText(desg);  editTextSalary.setText(sal);   } catch (JSONException e) {  e.printStackTrace();  }  }    private void updateEmployee(){  final String name = editTextName.getText().toString().trim();  final String desg = editTextDesg.getText().toString().trim();  final String salary = editTextSalary.getText().toString().trim();   class UpdateEmployee extends AsyncTask<Void,Void,String>{  ProgressDialog loading;  @Override  protected void onPreExecute() {  super.onPreExecute();  loading = ProgressDialog.*show*(ViewEmployee.this,"Updating...","Wait...",false,false);  }   @Override  protected void onPostExecute(String s) {  super.onPostExecute(s);  loading.dismiss();  Toast.*makeText*(ViewEmployee.this,s,Toast.*LENGTH\_LONG*).show();  }   @Override  protected String doInBackground(Void... params) {  HashMap<String,String> hashMap = new HashMap<>();  hashMap.put(Config.*KEY\_EMP\_ID*,id);  hashMap.put(Config.*KEY\_EMP\_NAME*,name);  hashMap.put(Config.*KEY\_EMP\_DESG*,desg);  hashMap.put(Config.*KEY\_EMP\_SAL*,salary);   RequestHandler rh = new RequestHandler();   String s = rh.sendPostRequest(Config.*URL\_UPDATE\_EMP*,hashMap);   return s;  }  }   UpdateEmployee ue = new UpdateEmployee();  ue.execute();  }   private void deleteEmployee(){  class DeleteEmployee extends AsyncTask<Void,Void,String> {  ProgressDialog loading;   @Override  protected void onPreExecute() {  super.onPreExecute();  loading = ProgressDialog.*show*(ViewEmployee.this, "Updating...", "Wait...", false, false);  }   @Override  protected void onPostExecute(String s) {  super.onPostExecute(s);  loading.dismiss();  Toast.*makeText*(ViewEmployee.this, s, Toast.*LENGTH\_LONG*).show();  }   @Override  protected String doInBackground(Void... params) {  RequestHandler rh = new RequestHandler();  String s = rh.sendGetRequestParam(Config.*URL\_DELETE\_EMP*, id);  return s;  }  }   DeleteEmployee de = new DeleteEmployee();  de.execute();  }   private void confirmDeleteEmployee(){  AlertDialog.Builder alertDialogBuilder = new AlertDialog.Builder(this);  alertDialogBuilder.setMessage("Are you sure you want to delete this employee?");   alertDialogBuilder.setPositiveButton("Yes",  new DialogInterface.OnClickListener() {  @Override  public void onClick(DialogInterface arg0, int arg1) {  deleteEmployee();  startActivity(new Intent(ViewEmployee.this,ViewAllEmployee.class));  }  });   alertDialogBuilder.setNegativeButton("No",  new DialogInterface.OnClickListener() {  @Override  public void onClick(DialogInterface arg0, int arg1) {   }  });   AlertDialog alertDialog = alertDialogBuilder.create();  alertDialog.show();  }   @Override  public void onClick(View v) {  if(v == buttonUpdate){  updateEmployee();  }   if(v == buttonDelete){  confirmDeleteEmployee();  }   }   } |

Activity\_main.xml

|  |
| --- |
| <LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"  xmlns:tools="http://schemas.android.com/tools" android:layout\_width="match\_parent"  android:orientation="vertical"  android:layout\_height="match\_parent" android:paddingLeft="@dimen/activity\_horizontal\_margin"  android:paddingRight="@dimen/activity\_horizontal\_margin"  android:paddingTop="@dimen/activity\_vertical\_margin"  android:paddingBottom="@dimen/activity\_vertical\_margin" tools:context=".MainActivity">   <TextView  android:layout\_width="wrap\_content"  android:layout\_height="wrap\_content"  android:text="username" />   <EditText  android:layout\_width="match\_parent"  android:layout\_height="wrap\_content"  android:id="@+id/editusername" />   <TextView  android:layout\_width="wrap\_content"  android:layout\_height="wrap\_content"  android:text="password" />   <EditText  android:layout\_width="match\_parent"  android:layout\_height="wrap\_content"   android:id="@+id/editpassword"  android:password="true" />     <Button  android:layout\_width="match\_parent"  android:layout\_height="wrap\_content"  android:text="Login"  android:id="@+id/buttonlogin" />  </LinearLayout> |

Activity\_view\_all\_employee

|  |
| --- |
| <LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"  xmlns:tools="http://schemas.android.com/tools" android:layout\_width="match\_parent"  android:orientation="vertical"  android:layout\_height="match\_parent" android:paddingLeft="@dimen/activity\_horizontal\_margin"  android:paddingRight="@dimen/activity\_horizontal\_margin"  android:paddingTop="@dimen/activity\_vertical\_margin"  android:paddingBottom="@dimen/activity\_vertical\_margin"  tools:context="net.simplifiedcoding.mysqlcrud.ViewAllEmployee">    <ListView  android:layout\_width="match\_parent"  android:layout\_height="wrap\_content"  android:id="@+id/listView" />   </LinearLayout> |

ViewEmployee.xml

|  |
| --- |
| <LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"  xmlns:tools="http://schemas.android.com/tools" android:layout\_width="match\_parent"  android:orientation="vertical"  android:layout\_height="match\_parent" android:paddingLeft="@dimen/activity\_horizontal\_margin"  android:paddingRight="@dimen/activity\_horizontal\_margin"  android:paddingTop="@dimen/activity\_vertical\_margin"  android:paddingBottom="@dimen/activity\_vertical\_margin"  tools:context="net.simplifiedcoding.mysqlcrud.ViewEmployee">   <TextView  android:layout\_width="wrap\_content"  android:layout\_height="wrap\_content"  android:text="Employee ID" />   <EditText  android:layout\_width="match\_parent"  android:layout\_height="wrap\_content"  android:enabled="false"  android:id="@+id/editTextId" />   <TextView  android:layout\_width="wrap\_content"  android:layout\_height="wrap\_content"  android:text="Employee Name" />   <EditText  android:layout\_width="match\_parent"  android:layout\_height="wrap\_content"  android:id="@+id/editTextName" />   <TextView  android:layout\_width="wrap\_content"  android:layout\_height="wrap\_content"  android:text="Designation" />   <EditText  android:layout\_width="match\_parent"  android:layout\_height="wrap\_content"  android:id="@+id/editTextDesg" />   <TextView  android:layout\_width="wrap\_content"  android:layout\_height="wrap\_content"  android:text="Salary" />   <EditText  android:layout\_width="match\_parent"  android:layout\_height="wrap\_content"  android:id="@+id/editTextSalary" />   <Button  android:layout\_width="match\_parent"  android:layout\_height="wrap\_content"  android:text="Update Employee"  android:id="@+id/buttonUpdate" />   <Button  android:layout\_width="match\_parent"  android:layout\_height="wrap\_content"  android:text="Delete Employee"  android:id="@+id/buttonDelete" />    </LinearLayout> |

List\_item.xml

|  |
| --- |
| <?xml version="1.0" encoding="utf-8"?> <LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"  android:orientation="vertical" android:layout\_width="match\_parent"  android:layout\_height="match\_parent">   <TextView  android:id="@+id/id"  android:layout\_width="wrap\_content"  android:layout\_height="wrap\_content" />   <TextView  android:id="@+id/name"  android:layout\_width="wrap\_content"  android:layout\_height="wrap\_content" />  </LinearLayout> |

Menu.xml

|  |
| --- |
| <LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"  xmlns:tools="http://schemas.android.com/tools" android:layout\_width="match\_parent"  android:orientation="vertical"  android:layout\_height="match\_parent" android:paddingLeft="@dimen/activity\_horizontal\_margin"  tools:context=".menu">   <TextView  android:layout\_width="wrap\_content"  android:layout\_height="wrap\_content"  android:text="Employee Name" />   <EditText  android:layout\_width="match\_parent"  android:layout\_height="wrap\_content"  android:id="@+id/editTextName" />   <TextView  android:layout\_width="wrap\_content"  android:layout\_height="wrap\_content"  android:text="Designation" />   <EditText  android:layout\_width="match\_parent"  android:layout\_height="wrap\_content"  android:id="@+id/editTextDesg" />   <TextView  android:layout\_width="wrap\_content"  android:layout\_height="wrap\_content"  android:text="Salary" />   <EditText  android:layout\_width="match\_parent"  android:layout\_height="wrap\_content"  android:id="@+id/editTextSalary" />   <Button  android:layout\_width="351dp"  android:layout\_height="wrap\_content"  android:text="Add Employee"  android:id="@+id/buttonAdd" />   <Button  android:layout\_width="351dp"  android:layout\_height="wrap\_content"  android:text="View Employee"  android:id="@+id/buttonView" /> </LinearLayout> |

