**Name : Qasim Mughal**

**Roll Num : 17271519-014**

Assignment Number 4  
**Mobile App with Rest Api using Retrofit Library**

[**https://github.com/qasimmughalz/Assignment**](https://github.com/qasimmughalz/Assignment)

**Login Api:**

public IEnumerable<UserLogin> Get()

{

AARDegreeEntities entities = new AARDegreeEntities();

return entities.UserLogins.ToList();

}

// GET api/<controller>/5

public UserLogin Get(int id)

{

AARDegreeEntities a = new AARDegreeEntities();

return a.UserLogins.FirstOrDefault(e => e.UserId == id);

}

public void Post([FromBody] UserLogin userLogin)

{

AARDegreeEntities degreeEntities = new AARDegreeEntities();

degreeEntities.UserLogins.Add(userLogin);

degreeEntities.SaveChanges();

}

}

**Declaring the fields**

**public class** Logindata {  
 **int UserId**;  
 String **UserName**;  
 String **UserPassword**;  
 String **UserType**;  
  
 **public** Logindata(**int** userId, String userName, String userPassword, String userType) {  
 **UserId** = userId;  
 **UserName** = userName;  
 **UserPassword** = userPassword;  
 **UserType** = userType;  
 }  
  
 **public int** getUserId() {  
 **return UserId**;  
 }  
  
 **public** String getUserName() {  
 **return UserName**;  
 }  
  
 **public** String getUserPassword() {  
 **return UserPassword**;  
 }  
  
 **public** String getUserType() {  
 **return UserType**;  
 }  
}

**package** com.example.retroApp;  
  
**public class** StudentData {  
 **int Userid**;  
 **public** String **UserName**;  
 String **UserLastName**;  
 String **Cnic**;  
 String **Adress**;  
 String **Degree**;  
 String **Subject**;  
 String **PassingYear**;  
 String **Comment**;  
 String **Status**;  
  
 **public** StudentData(**int** userid, String userName, String userLastName, String cnic, String adress, String degree, String subject, String passingYear, String comment, String status) {  
 **Userid** = userid;  
 **UserName** = userName;  
 **UserLastName** = userLastName;  
 **Cnic** = cnic;  
 **Adress** = adress;  
 **Degree** = degree;  
 **Subject** = subject;  
 **PassingYear** = passingYear;  
 **Comment** = comment;  
 **Status** = status;  
 }  
  
 **public int** getUserid() {  
 **return Userid**;  
 }  
  
 **public** String getUserName() {  
 **return UserName**;  
 }  
  
 **public** String getUserLastName() {  
 **return UserLastName**;  
 }  
  
 **public** String getCnic() {  
 **return Cnic**;  
 }  
  
 **public** String getAdress() {  
 **return Adress**;  
 }  
  
 **public** String getDegree() {  
 **return Degree**;  
 }  
  
 **public** String getSubject() {  
 **return Subject**;  
 }  
  
 **public** String getPassingYear() {  
 **return PassingYear**;  
 }  
  
 **public** String getComment() {  
 **return Comment**;  
 }  
  
 **public** String getStatus() {  
 **return Status**;  
 }  
}

**Here we will declare interface**

**package** com.example.retroApp;  
  
**import** java.util.List;  
  
**import** retrofit2.Call;  
**import** retrofit2.http.Field;  
**import** retrofit2.http.GET;  
**import** retrofit2.http.POST;  
  
**public interface** LoginnApiI {  
 @GET(**"api/login"**)  
 Call<List<Logindata>> getAllStudentLogin();  
 @POST(**"api/login"**)  
 Call<Logindata> sendpost2(  
 @Field(**"Id"**)**int** id,  
 @Field(**"name"**)String title,  
 @Field(**"password"**)String text,  
 @Field(**"type"**)String type  
 );  
}

**package** com.example.retroApp;  
  
**import** java.util.List;  
  
**import** retrofit2.Call;  
**import** retrofit2.http.Field;  
**import** retrofit2.http.GET;  
**import** retrofit2.http.POST;  
**import** retrofit2.http.Url;  
  
**public interface** StudentdataI {  
@GET(**"api/usersdata"**)  
Call<List<StudentData>> getAllStudentDegreeRequest();  
 @POST(**"api/usersdata"**)  
 Call<StudentData> sendpost2(  
 @Field(**"Id"**)**int** id,  
 @Field(**"name"**)String title,  
 @Field(**"lastname"**)String lname,  
 @Field(**"type"**)String type,  
 @Field(**"date"**)String date,  
 @Field(**"cinc"**)String cinc,  
 @Field(**"Adress"**)String Adress,  
 @Field(**"Degree"**)String Degree,  
 @Field(**"Subject"**)String Subject,  
 @Field(**"Comment"**)String Comment,  
 @Field(**"Status"**)String Status  
  
  
 );  
 @GET  
 Call<List<StudentData>> getDelete(@Url String url);  
 Call<StudentData> delete(  
 @Field(**"Id"**)**int** id);  
}

**void** LoginSetretrofit()  
 {  
 Retrofit retrofit=**new** Retrofit.Builder()  
 .baseUrl(**"https://10.106.246.118:44359/"**)  
 .addConverterFactory(GsonConverterFactory.*create*() )  
 .build();  
 LoginnApiI api=retrofit.create(LoginnApiI.**class**);  
 Call<List<Logindata>> listCall=api.getAllStudentLogin();  
 *// to execute in other thread....* listCall.enqueue(**new** Callback<List<Logindata>>() {  
 @Override  
 **public void** onResponse(Call<List<Logindata>> call, Response<List<Logindata>> response) {  
 **if** ((!response.isSuccessful())) {  
 Log.*i*(**"Size1"**, String.*valueOf*(response.message()));  
 **return**;  
 }  
 List<Logindata> postList = response.body();  
  
 Log.*i*(**"Size2"**, String.*valueOf*(postList.size()));  
 **for** (Logindata p : postList) {  
 {  
 **try** {  
 *studentlogINList*.add(**new** Logindata(p.getUserId(),p.getUserName(),p.getUserPassword(),p.getUserType()));  
 } **catch** (Exception e) {  
 e.getMessage();  
 }  
 }  
 }  
 }  
  
 @Override  
 **public void** onFailure(Call<List<Logindata>> call, Throwable t) {  
 Log.*i*(**"Size4"**,String.*valueOf*(t.getMessage()));  
 }  
 });  
 }  
  
 **void** send\_toServer(**int** id ,String name,String password,String Type){  
 Logindata post1=**new** Logindata(id, name,password,Type);  
 LoginnApiI api=**retrofit**.create(LoginnApiI.**class**);  
 Call<Logindata> postCall=api.sendpost2(id, name,password,Type);  
 postCall.enqueue(**new** Callback<Logindata>() {  
 @Override  
 **public void** onResponse(Call<Logindata> call, Response<Logindata> response) {  
 **if**(!(response.isSuccessful()))  
 {  
 **return**;  
 }  
 Logindata p=response.body();  
  
  
  
 }  
  
 @Override  
 **public void** onFailure(Call<Logindata> call, Throwable t) {  
  
 }  
 });  
 }  
 **void** send\_toServerForDelete(**int** id )  
 {  
  
 StudentdataI api=**retrofit**.create(StudentdataI.**class**);  
 Call<StudentData> postCall=api.delete(id);  
 postCall.enqueue(**new** Callback<StudentData>() {  
 @Override  
 **public void** onResponse(Call<StudentData> call, Response<StudentData> response) {  
 **if**(!(response.isSuccessful()))  
 {  
 **return**;  
 }  
 StudentData p=response.body();  
  
  
  
 }  
  
 @Override  
 **public void** onFailure(Call<StudentData> call, Throwable t) {  
  
 }  
 });  
 }  
  
  
  
 **void** send\_toServerForupdate(**int** id )  
 {  
  
 StudentdataI api=**retrofit**.create(StudentdataI.**class**);  
 Call<StudentData> postCall=api.delete(id);  
 postCall.enqueue(**new** Callback<StudentData>() {  
 @Override  
 **public void** onResponse(Call<StudentData> call, Response<StudentData> response) {  
 **if**(!(response.isSuccessful()))  
 {  
 **return**;  
 }  
 StudentData p=response.body();  
  
  
  
 }  
  
 @Override  
 **public void** onFailure(Call<StudentData> call, Throwable t) {  
  
 }  
 });  
 }  
  
  
  
}  
 // GET api/<controller>

public IEnumerable<Userdata> Get()

{

AARDegreeEntities entities = new AARDegreeEntities();

return entities.Userdatas.ToList();

}

// GET api/<controller>/5

public Userdata Get(int id)

{

AARDegreeEntities a = new AARDegreeEntities();

return a.Userdatas.FirstOrDefault(e => e.Userid == id);

}

// GET api/

public void Post([FromBody] Userdata u)

{

AARDegreeEntities degreeEntities = new AARDegreeEntities();

degreeEntities.Userdatas.Add(u);

degreeEntities.SaveChanges();

}

public void Delete(int id)

{

AARDegreeEntities a = new AARDegreeEntities();

a.Userdatas.Remove(a.Userdatas.FirstOrDefault(e => e.Userid == id));

}

public void Put(int id, [FromBody] Userdata u)

{

AARDegreeEntities a = new AARDegreeEntities();

var v = a.Userdatas.FirstOrDefault(e => e.Userid == id);

v.Userid = u.Userid;

v.UserLastName = u.UserLastName;

v.Subject = u.Subject;

v.Status = u.Status;

v.PassingYear = u.PassingYear;

v.Degree = u.Degree;

a.SaveChanges();

}

}

}