WEB_ICP10

DHARMA TEJA K

Email: dkbmy@umsystem.edu

GitHub Link: https://github.com/Dharmateja183/Web-and-Mobile-programming-spring-2022/tree/main/mobile%20part/ICP10

Avinash Reddy T

Email: atfkh@umsystem.edu

GitHub link: https://github.com/avinashreddy3/WebDevCourse/tree/main/MobilePart/ICP10

In Class Programming:

In this ICP, we are going to utilize android studio, java programming, xml languages for mobile application development, to print the GitHub users 'ID' and 'User Name' data from the website, https://api.github.com.

1) When the 'Run' button is pressed on an Android phone, the information about GitHub users will be printed.
2) The data will be scrolled to display all GitHub users' data.

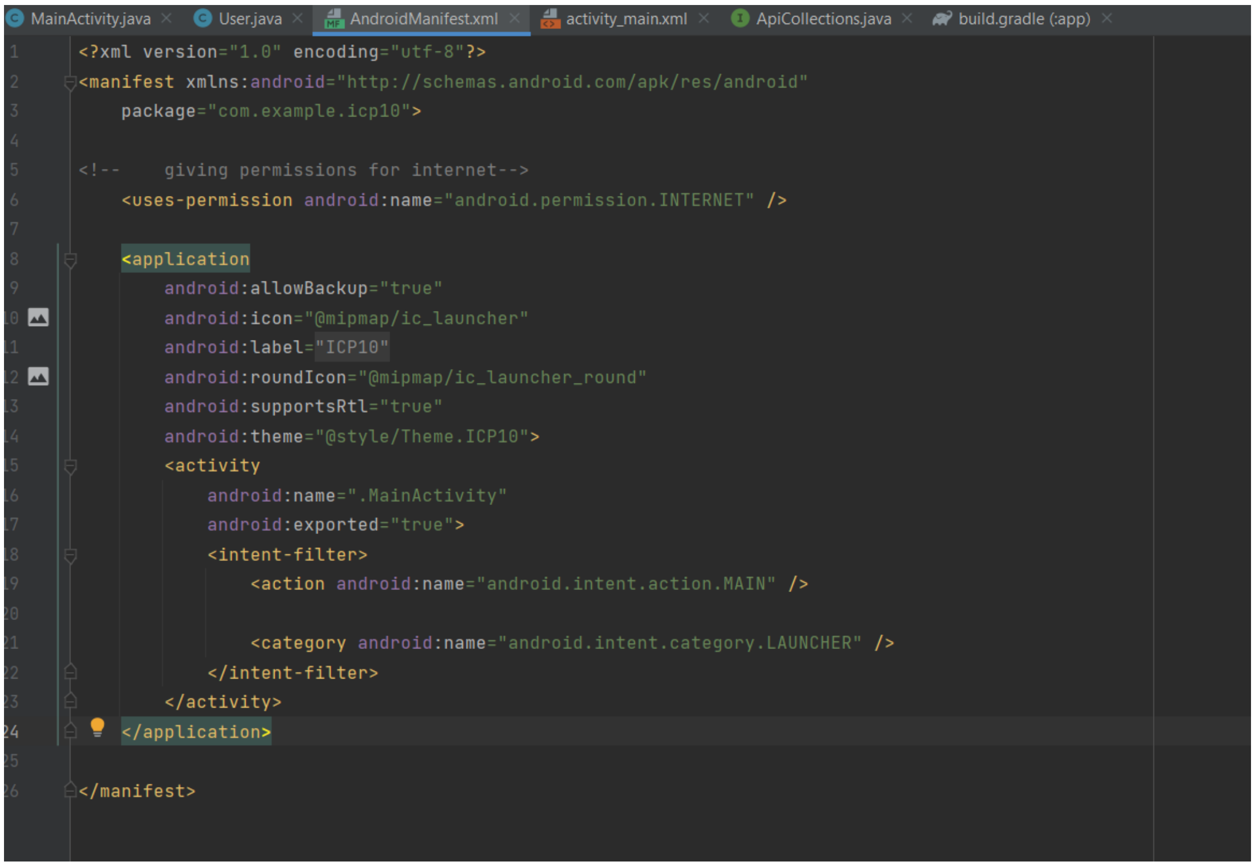
Procedure:

First we have created a new project with empty activity using android studio application.

This provides .java, .xml files which are used to display the user's details of github.

1. Adding the internet permission in android manifest.xml file.

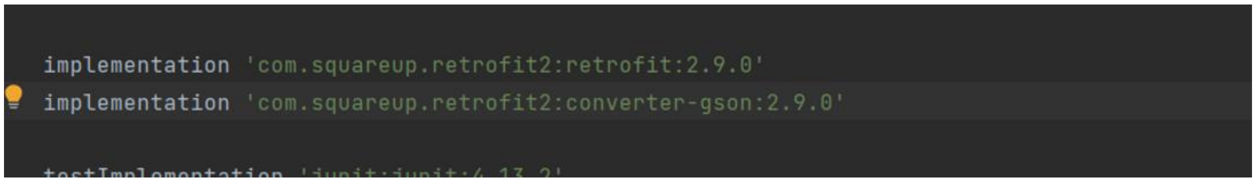
We must add an extra permission to androidManifest.xml in order to gain access to the internet; otherwise, we will be unable to retrieve information from the internet.



```xml
<?xml version="1.0" encoding="utf-8"?>
<manifest xmlns:android="http://schemas.android.com/apk/res/android"
    package="com.example.icp10">

<!--    giving permissions for internet-->
    <uses-permission android:name="android.permission.INTERNET" />

    <application
        android:allowBackup="true"
        android:icon="@mipmap/ic_launcher"
        android:label="ICP10"
        android:roundIcon="@mipmap/ic_launcher_round"
        android:supportsRtl="true"
        android:theme="@style/Theme.ICP10">
        <activity
            android:name=".MainActivity"
            android:exported="true">
            <intent-filter>
                <action android:name="android.intent.action.MAIN" />

                <category android:name="android.intent.category.LAUNCHER" />
            </intent-filter>
        </activity>
    </application>

</manifest>
```

2. Add few other dependencies to the gradlebuild file.

Your HTTP API becomes a Java interface using Retrofit.



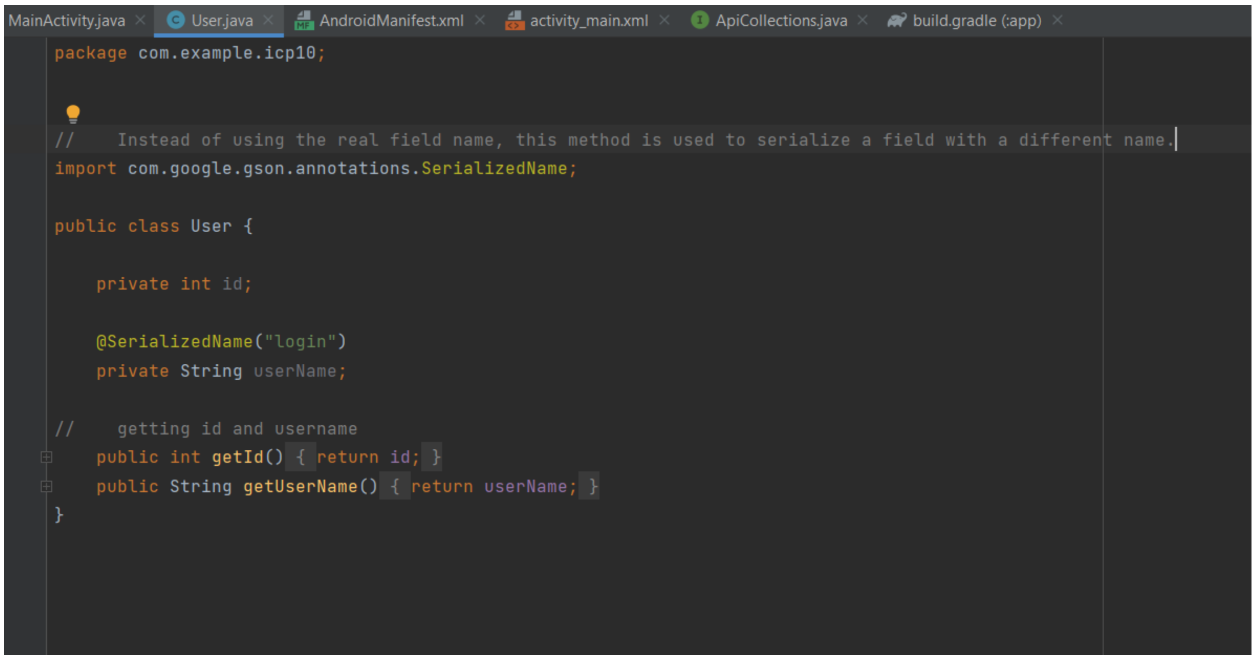
```
implementation 'com.squareup.retrofit2:retrofit:2.9.0'
implementation 'com.squareup.retrofit2:converter-gson:2.9.0'

testImplementation 'junit:junit:4.13.2'
```

3. **Creating user.java class.**

Here, create a user.java class with two variables named “ID” and “username” as shown in below screenshot.

And we also use serializedName annotation,

This method is used to serialize a field with a different name instead of using the real field name.

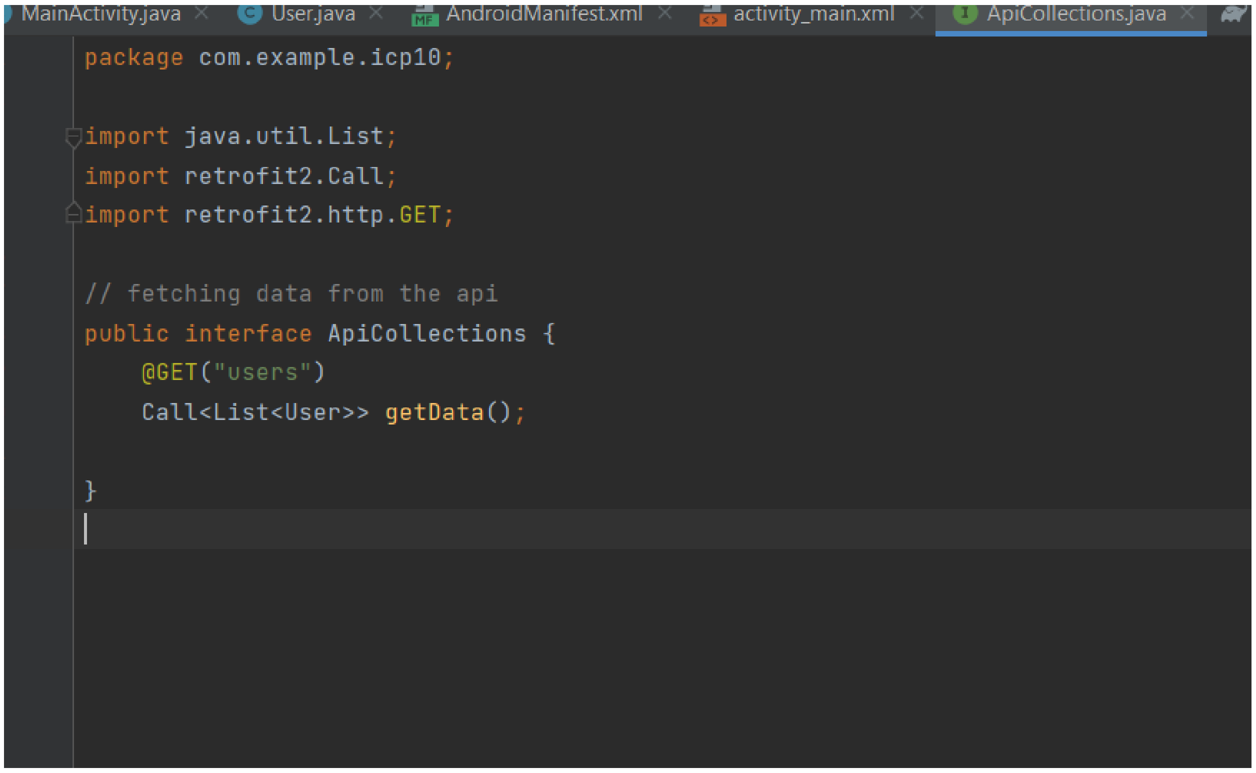


```
package com.example.icp10;


//    Instead of using the real field name, this method is used to serialize a field with a different name.
import com.google.gson.annotations.SerializedName;

public class User {

    private int id;

    @SerializedName("login")
    private String userName;

//    getting id and username
    public int getId() { return id; }
    public String getUserName() { return userName; }
}
```

4. **Creating ApiCollection.java file.**

Here, we create a java class named ApiCollection and selected it as an interface for API call.

And we get the required details by making GET request and we call it with JSON object which is named as getData() method.



```java
package com.example.icp10;

import java.util.List;
import retrofit2.Call;
import retrofit2.http.GET;

// fetching data from the api
public interface ApiCollections {
    @GET("users")
    Call<List<User>> getData();

}
```

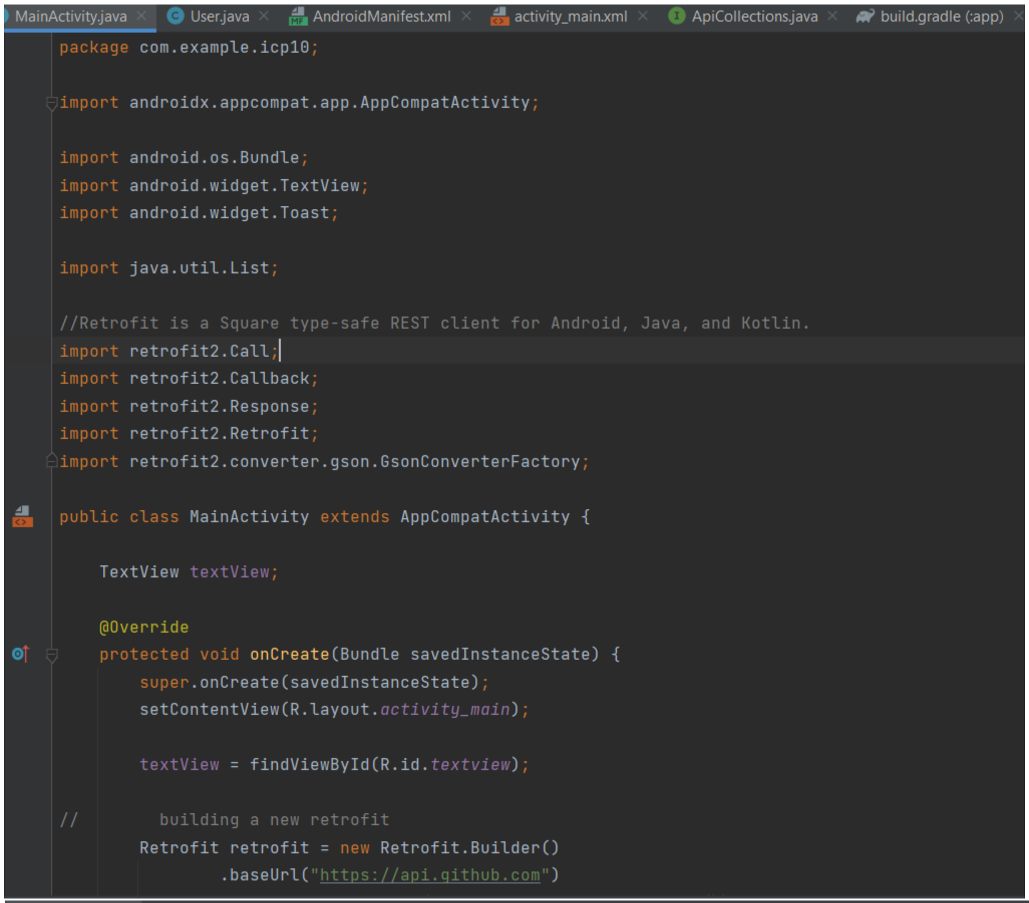
5. Mainactivity.java file

Here we used REST api to display the information we get from the api's.

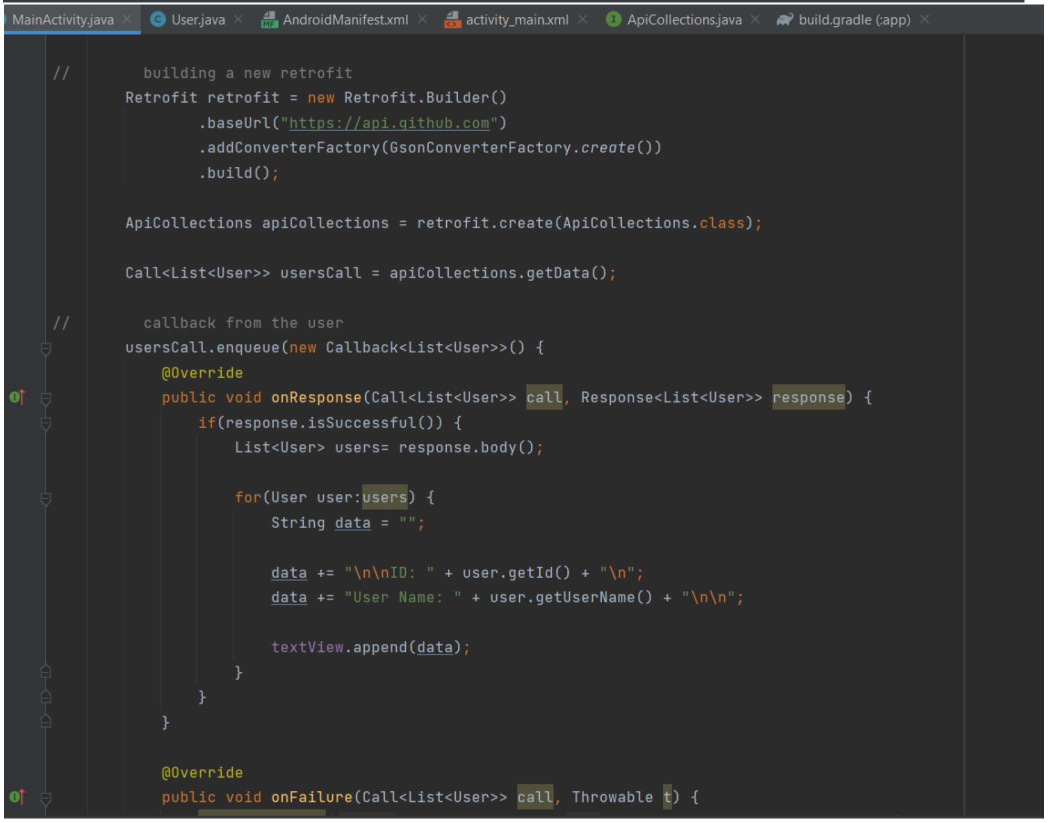
And we import the Retrofit libraries as well. Retrofit is a square type-safe REST client for android, java and kotlin.

We create a new Retrofit and pass the github url link to the Retrofit Builder().

Below are the snaps of mainactivity.java file.



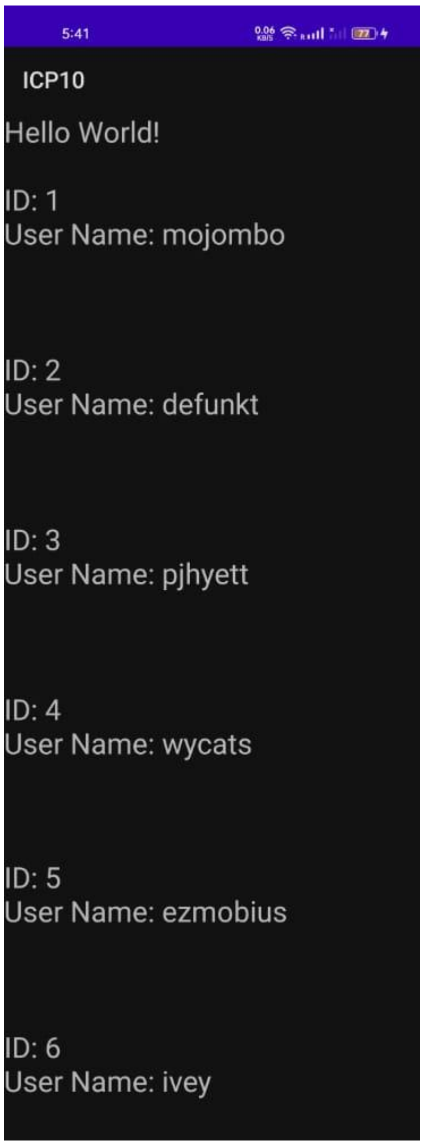
```java
package com.example.icp10;

import androidx.appcompat.app.AppCompatActivity;

import android.os.Bundle;
import android.widget.TextView;
import android.widget.Toast;

import java.util.List;

//Retrofit is a Square type-safe REST client for Android, Java, and Kotlin.
import retrofit2.Call;
import retrofit2.Callback;
import retrofit2.Response;
import retrofit2.Retrofit;
import retrofit2.converter.gson.GsonConverterFactory;

public class MainActivity extends AppCompatActivity {

    TextView textView;

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

        textView = findViewById(R.id.textview);

//        building a new retrofit
        Retrofit retrofit = new Retrofit.Builder()
                .baseUrl("https://api.github.com")
```



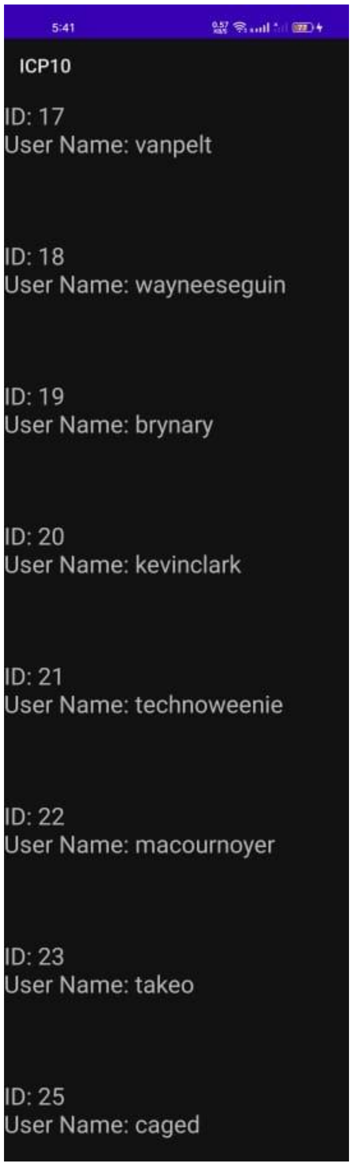
```java
//        building a new retrofit
        Retrofit retrofit = new Retrofit.Builder()
                .baseUrl("https://api.github.com")
                .addConverterFactory(GsonConverterFactory.create())
                .build();

        ApiCollections apiCollections = retrofit.create(ApiCollections.class);

        Call<List<User>> usersCall = apiCollections.getData();

//        callback from the user
        usersCall.enqueue(new Callback<List<User>>() {
            @Override
            public void onResponse(Call<List<User>> call, Response<List<User>> response) {
                if(response.isSuccessful()) {
                    List<User> users= response.body();

                    for(User user:users) {
                        String data = "";

                        data += "\n\nID: " + user.getId() + "\n";
                        data += "User Name: " + user.getUserName() + "\n\n";

                        textView.append(data);
                    }
                }
            }

            @Override
            public void onFailure(Call<List<User>> call, Throwable t) {
```

Outputs:

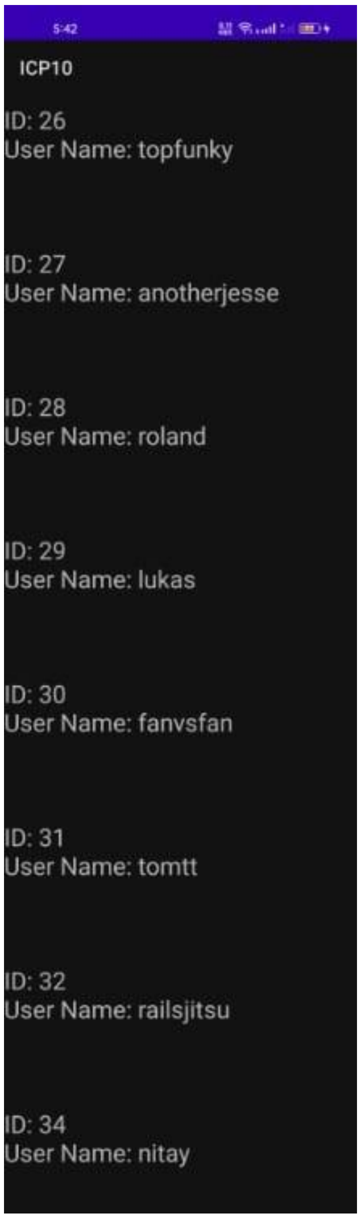
Below are the output snippets when we run the application on any android device.



```
ICP10
Hello World!
ID: 1
User Name: mojombo
ID: 2
User Name: defunkt
ID: 3
User Name: pjhyett
ID: 4
User Name: wycats
ID: 5
User Name: ezmobius
ID: 6
User Name: ivey
```



```
ICP10
ID: 17
User Name: vanpelt
ID: 18
User Name: wayneeseguin
ID: 19
User Name: brynary
ID: 20
User Name: kevinclark
ID: 21
User Name: technoweenie
ID: 22
User Name: macournoyer
ID: 23
User Name: takeo
ID: 25
User Name: caged
```



```
ICP10
ID: 26
User Name: topfunky
ID: 27
User Name: anotherjesse
ID: 28
User Name: roland
ID: 29
User Name: lukas
ID: 30
User Name: fanvsfan
ID: 31
User Name: tomtt
ID: 32
User Name: railsjitsu
ID: 34
User Name: nitay
```