Terna Engineering College

Computer Engineering Department

Program: Sem VII

Course: MOBILE COMMUNICATION & COMPUTING AND MOBILE APPLICATION DEVELOPMENT LAB (MCC & MAD Lab)

Experiment No. 09

PART B

(PART B: TO BE COMPLETED BY STUDENTS)

(Students must submit the soft copy as per the following segments within two hours of the practical. The soft copy must be uploaded on the Blackboard or emailed to the concerned lab in charge faculties at the end of the practical in case there is no Blackboard access available)

Roll No. 50	Name: AMEY THAKUR
Class: BE-COMPS-50	Batch: B3
Date of Experiment: 01-10-2021	Date of Submission: 01-10-2021
Grade :	

Aim: Write a program for android phone display user form and its validation for android phones.

B.1 Software Code written by a student:

Open build.gradle file in Gradle Scripts and add new library as below:

```
implementation 'com.mobsandgeeks:android-saripaar:2.0.3'
```

Open res\values\strings.xml file and add new string as below:

```
<resources>
        <string name="app_name">Learn Android with Real Apps</string>
        <string name="i_agree">I agree to the terms of use</string>
        <string name="username">Username</string>
        <string name="password">Password</string>
        <string name="confirm_password">Confirm Password</string>
        <string name="email">Email</string>
        <string name="website">Website</string>
        <string name="age">Age</string>
        <string name="phone">Phone</string>
        <string name="save">Save</string>
        <string name="username_already_exists">Username already exists</string>
</resources>
```

Open res\layout\activity_main.xml file and create layout as below:

```
<?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=".MainActivity">
<EditText
        android:id="@+id/editTextUsername"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:ems="10"
        android:hint="@string/username"
        android:inputType="textPersonName" />
<EditText
        android:id="@+id/editTextPassword"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:ems="10"
        android:hint="@string/password"
        android:inputType="textPassword" />
<EditText
        android:id="@+id/editTextConfirmPassword"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:ems="10"
        android:hint="@string/confirm_password"
        android:inputType="textPassword" />
<EditText
        android:id="@+id/editTextPhone"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:ems="10"
        android:hint="@string/phone"
        android:inputType="phone" />
<EditText
        android:id="@+id/editTextAge"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:ems="10"
        android:hint="@string/age"
        android:inputType="number" />
<EditText
        android:id="@+id/editTextEmail"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:ems="10"
        android:hint="@string/email"
        android:inputType="textEmailAddress" />
<EditText
        android:id="@+id/editTextWebsite"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:ems="10"
        android:hint="@string/website"
        android:inputType="textPersonName" />
<CheckBox
        android:id="@+id/checkBoxAgree"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:text="@string/i_agree" />
<Button
        android:id="@+id/buttonSave"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:text="@string/save" />
</LinearLayout>
```

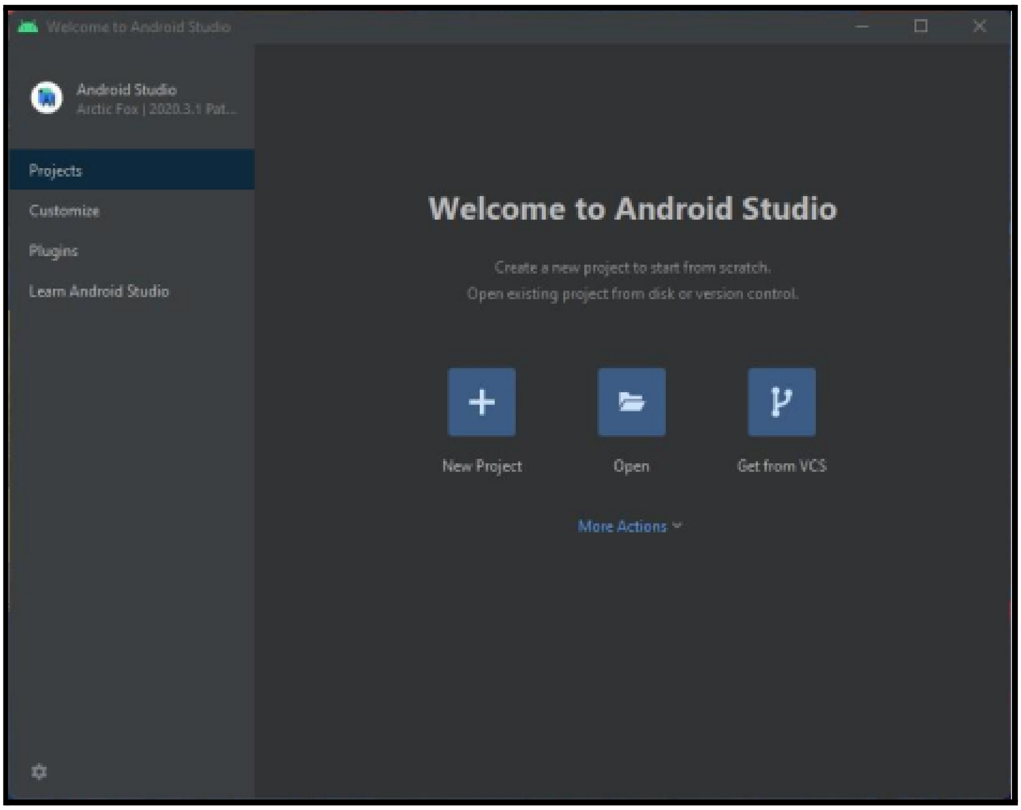
Add code to MainActivity.java in android.demo.learnandroidwithrealapps package as below:

```
package android.demo.learnandroidwithrealapps;
import android.os.Bundle;
import android.support.v7.app.AppCompatActivity;
import android.view.View;
import android.widget.Button;
import android.widget.CheckBox;
import android.widget.EditText;
import android.widget.Toast;
import com.mobsandgeeks.saripaar.ValidationError;
import com.mobsandgeeks.saripaar.Validator;
import com.mobsandgeeks.saripaar.annotation.Checked;
import com.mobsandgeeks.saripaar.annotation.ConfirmPassword;
import com.mobsandgeeks.saripaar.annotation.Email;
import com.mobsandgeeks.saripaar.annotation.Length;
import com.mobsandgeeks.saripaar.annotation.Max;
import com.mobsandgeeks.saripaar.annotation.Min;
import com.mobsandgeeks.saripaar.annotation.NotEmpty;
import com.mobsandgeeks.saripaar.annotation.Password;
import com.mobsandgeeks.saripaar.annotation.Pattern;
import com.mobsandgeeks.saripaar.annotation.Url;
import java.util.List;
public class MainActivity extends AppCompatActivity implements
Validator.ValidationListener {
        @NotEmpty
        @Length(min = 3, max = 10)
        private EditText editTextUsername;
        @NotEmpty
        @Password
        @Pattern(regex = "((?=.*\\d)(?=.*[a-z])(?=.*[A-Z])(?=.*[@#$%]).{6,20})") private
        EditText editTextPassword;
        @ConfirmPassword
        private EditText editTextConfirmPassword;
        @NotEmpty
        @Pattern(regex     =     "^\\(?([0-9]{3})\\)?[-.\\s]?([0-9]{3})[-.\\s]?([0-9]{4})$")
        private EditText editTextPhone;
        @Min(18)
        @Max(120)
        private EditText editTextAge;
        @NotEmpty
        @Email
        private EditText editTextEmail;
        @Url
        private EditText editTextWebsite;
        @Checked
        private CheckBox checkBoxAgree;
        private Button buttonSave;
        private Validator validator;

@Override
protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);
        initView();
        validator = new Validator(this);
        validator.setValidationListener(this);
}
private void initView() {
        editTextUsername = findViewById(R.id.editTextUsername);
        editTextPassword = findViewById(R.id.editTextPassword);
        editTextConfirmPassword = findViewById(R.id.editTextConfirmPassword);
        editTextAge = findViewById(R.id.editTextAge);
        editTextPhone = findViewById(R.id.editTextPhone);
        editTextEmail = findViewById(R.id.editTextEmail);
        editTextWebsite = findViewById(R.id.editTextWebsite);
        checkBoxAgree = findViewById(R.id.checkBoxAgree);
        buttonSave = findViewById(R.id.buttonSave);
        buttonSave.setOnClickListener(new View.OnClickListener() {
                @Override
                public void onClick(View view) {
                buttonSave_onClick(view);
                }
        });
}
private void buttonSave_onClick(View view) {
        validator.validate();
        String username = editTextUsername.getText().toString();
        if (username.equalsIgnoreCase("pmk")) {
                editTextUsername.setError(getText(R.string.username_already_exist
        s)); }
}
@Override
public void onValidationSucceeded() {
        Toast.makeText(this, "We got it right!", Toast.LENGTH_SHORT).show();
}
@Override
public void onValidationFailed(List<ValidationError> errors) {
        for (ValidationError error : errors) {
                View view = error.getView();
                String message = error.getCollatedErrorMessage(this);
                // Display error messages
                if (view instanceof EditText) {
                        ((EditText) view).setError(message);
                } else {
                        Toast.makeText(this, message, Toast.LENGTH_LONG).show();
                        }
                }
        }
}
```

B.2 Input and Output:

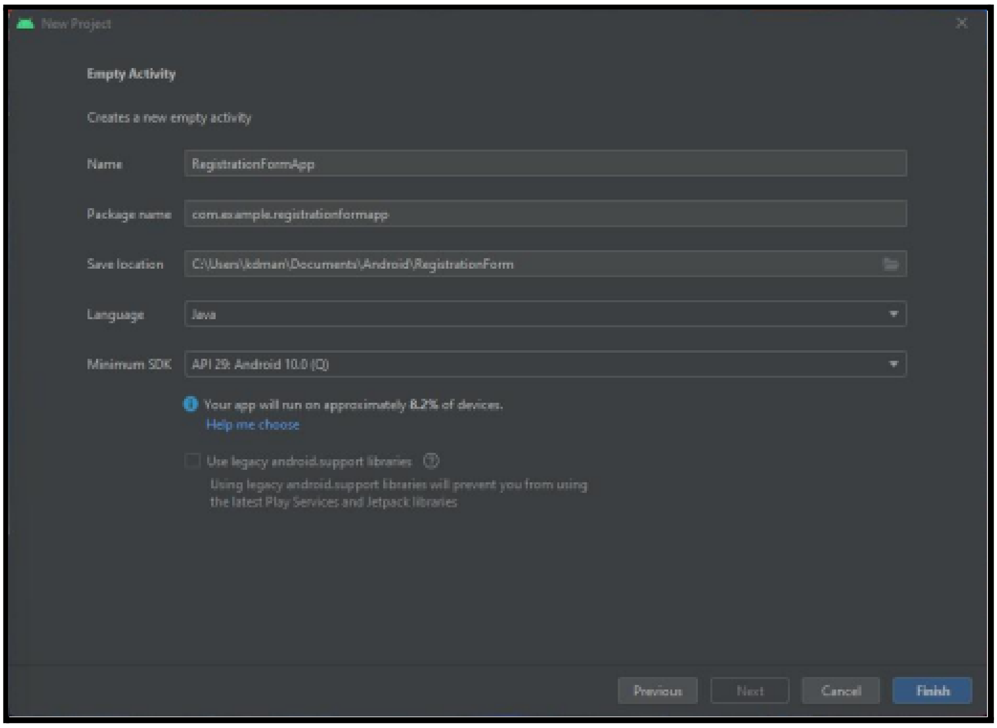
- ➔ Step 1: Input Project Name and Select Project Location



Welcome to Android Studio

Create a new project to start from scratch.
Open existing project from disk or version control.

- ➔ Step 2: Select SDK for Android App and name your project and click on "finish"



Empty Activity

Creates a new empty activity

Name: RegistrationFormApp

Package name: com.example.registrationformapp

Save location: C:\Users\kdman\Documents\Android\RegistrationForm

Language: Java

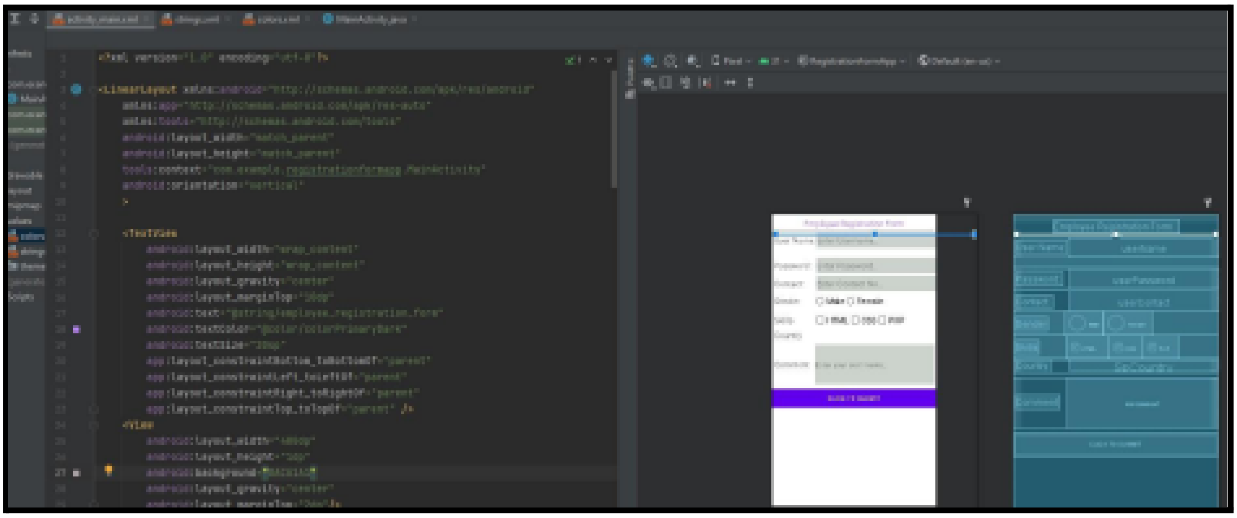
Minimum SDK: API 29: Android 10.0 (Q)

Your app will run on approximately 8.2% of devices.
Help me choose

□ Use legacy android.support libraries

Using legacy android.support libraries will prevent you from using the latest Play Services and Jetpack libraries

➔ Step 3: Select Default Activity for App

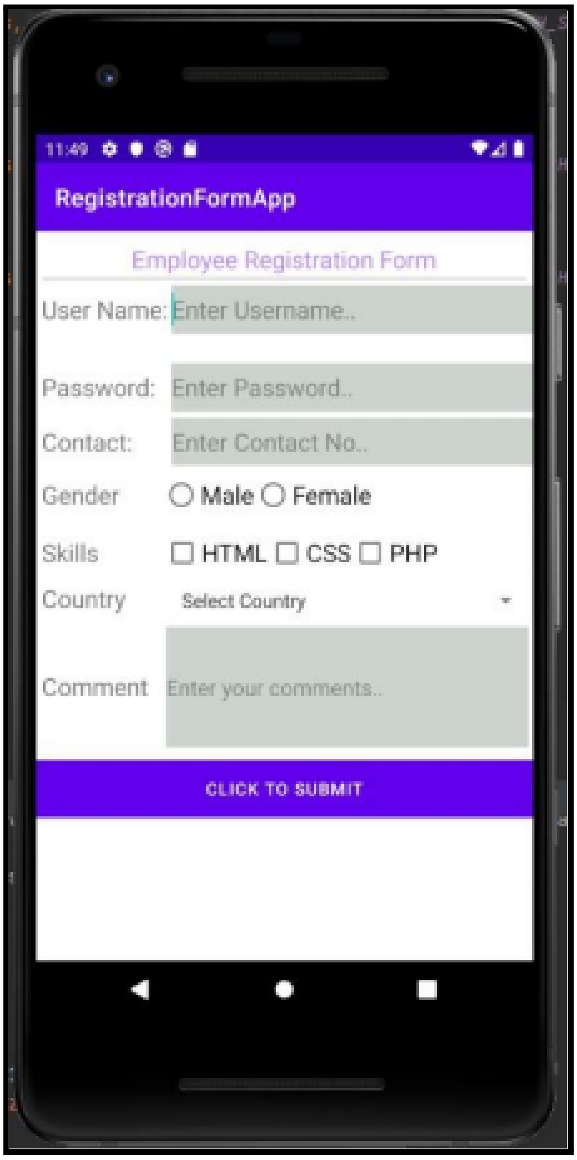


```
<?xml version="1.0" encoding="utf-8"?>

<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"
    tools:context="com.example.registrationformapp.MainActivity"
    android:orientation="vertical"
    >

    <TextView
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_gravity="center"
        android:layout_marginTop="10dp"
        android:text="@string/employee_registration_form"
        android:textColor="@color/colorPrimaryDark"
        android:textSize="20dp"
        app:layout_constraintBottom_toBottomOf="parent"
        app:layout_constraintLeft_toLeftOf="parent"
        app:layout_constraintRight_toRightOf="parent"
        app:layout_constraintTop_toTopOf="parent" />
    <View
        android:layout_width="400dp"
        android:layout_height="2dp"
        android:background="#[illegible]"
        android:layout_gravity="center"
        android:layout_marginTop="2dp"/>
```

➔ Output:



11:49

RegistrationFormApp

Employee Registration Form

User Name: Enter Username..

Password: Enter Password..

Contact: Enter Contact No..

Gender ○ Male ○ Female

Skills □ HTML □ CSS □ PHP

Country Select Country

Comment Enter your comments..

CLICK TO SUBMIT

B.3 Observations and learning:

After successful completion of this experiment students will be able to create a user form and its validation.

B.4 Conclusion:

We have successfully a program for android phones to display user form and its validation for android phones.

B.5 Question of Curiosity

1. What are the Android app components?

Ans:

- ➔ Activities: They dictate the UI and handle the user interaction to the smartphone screen.
- ➔ Services: They handle background processing associated with an application.
- ➔ Broadcast Receivers: They handle communication between Android OS and applications.
- ➔ Content Providers: They handle data and database management issues.

2. Explain in detail android run time components

Ans:

- ➔ Android currently uses Android Runtime (ART) to execute application code. ART is preceded by the Dalvik Runtime that compiled developer code to Dalvik Executable files (Dex files). These execution environments are optimized for the android platform taking into consideration the processor and memory constraints on mobile devices.
- ➔ The runtime translates code written by programmers into machine code that does computations and utilizes android framework components to deliver functionality. Android hosts multiple applications and system components that each run in their processes.