Android

Table of Contents

[Layouts 3](#_Toc33386055)

[View Adapter 3](#_Toc33386056)

[List Adapter 3](#_Toc33386057)

[activity\_main.xml 3](#_Toc33386058)

[simple\_list\_item\_1.xml 3](#_Toc33386059)

[MainActivity.java 3](#_Toc33386060)

[Custom List Adapter 3](#_Toc33386061)

[activity\_main.xml 3](#_Toc33386062)

[simple\_list\_item\_1.xml 4](#_Toc33386063)

[User.java 4](#_Toc33386064)

[UserAdapter.java 5](#_Toc33386065)

[MainActivity.java 5](#_Toc33386066)

[Grid 6](#_Toc33386067)

[activity\_main.xml 6](#_Toc33386068)

[my\_adaptor.java 6](#_Toc33386069)

[MainActivity.java 7](#_Toc33386070)

[Controls / Widgets 8](#_Toc33386071)

[Progress Bar 8](#_Toc33386072)

[MainActivity.java 8](#_Toc33386073)

[Alert Dialog 9](#_Toc33386074)

[myDialog.java 9](#_Toc33386075)

[MainActivity.java 9](#_Toc33386076)

[Date Picker 10](#_Toc33386077)

[activity\_main.xml 10](#_Toc33386078)

[MainActivity.java 11](#_Toc33386079)

[Time Picker 13](#_Toc33386080)

[activity\_main.xml 13](#_Toc33386081)

[MainActivity.java 14](#_Toc33386082)

[Activity 16](#_Toc33386083)

[Moving to next activity 16](#_Toc33386084)

[Passing data to Activity 16](#_Toc33386085)

[Passing Extras 16](#_Toc33386086)

[Passing Bundle 17](#_Toc33386087)

[Passing Parcable 18](#_Toc33386088)

[Return Data from Activity on Back 19](#_Toc33386089)

[Activity 1 20](#_Toc33386090)

[Activity 2 20](#_Toc33386091)

# Layouts

* Linear – Stack
* Table
* Grid
* Relative
* Constraint

# View Adapter

## List Adapter

### activity\_main.xml

<?xml version="1.0" encoding="utf-8"?>  
<android.support.constraint.ConstraintLayout  
 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.jfedin.listviewexample.MainActivity">  
  
 <ListView  
 android:id="@+id/myListView"  
 android:layout\_width="0dp"  
 android:layout\_height="match\_parent" />  
</android.support.constraint.ConstraintLayout>

### simple\_list\_item\_1.xml

<?xml version="1.0" encoding="utf-8"?>  
<TextView xmlns:android="http://schemas.android.com/apk/res/android"  
 android:layout\_width="match\_parent"  
 android:layout\_height="match\_parent"></TextView>

### MainActivity.java

String[] myStringArray = getResources().getStringArray(R.array.my\_string\_array);  
ArrayAdapter<String> adapter = new ArrayAdapter<String>(this, android.R.layout.simple\_list\_item\_1, myStringArray);  
  
ListView listView = (ListView) findViewById(R.id.myListView);  
listView.setAdapter(adapter);

## Custom List Adapter

### activity\_main.xml

<?xml version="1.0" encoding="utf-8"?>  
<android.support.constraint.ConstraintLayout  
 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.jfedin.listviewexample.MainActivity">  
  
 <ListView  
 android:id="@+id/myListView"  
 android:layout\_width="0dp"  
 android:layout\_height="match\_parent" />  
</android.support.constraint.ConstraintLayout>

### simple\_list\_item\_1.xml

<?xml version="1.0" encoding="utf-8"?>  
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"  
 android:layout\_width="match\_parent"  
 android:layout\_height="match\_parent"  
 android:padding="5dp">  
  
 <ImageView  
 android:id="@+id/logo"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:layout\_marginLeft="5px"  
 android:layout\_marginRight="20px"  
 android:layout\_marginTop="5px"></ImageView>  
  
 <TextView  
 android:id="@+id/label"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:layout\_margin="25dp"  
 android:padding="25dp"  
 android:text="@+id/label"  
 android:textSize="30px"></TextView>  
  
 <TextView  
 android:id="@+id/label2"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:layout\_margin="25dp"  
 android:padding="25dp"  
 android:text="@+id/label"  
 android:textSize="30px"></TextView>  
</LinearLayout>

### User.java

public class User {  
 public String firstName;  
 public String lastName;  
  
 public User(String firstName, String lastName) {  
 this.firstName = firstName;  
 this.lastName = lastName;  
 }  
}

### UserAdapter.java

public class UserAdapter extends ArrayAdapter<User> {  
 public UserAdapter(Context context, ArrayList<User> users) {  
 super(context, 0, users);  
 }  
  
 @Override  
 public View getView(int position, View convertView, ViewGroup parent) {  
 User user = getItem(position);  
  
 if (convertView == null) {  
 convertView = LayoutInflater.from(getContext()).inflate(R.layout.simple\_list\_item\_1, parent, false);  
 }  
  
 ImageView image = (ImageView) convertView.findViewById(R.id.logo);  
 TextView text1 = (TextView) convertView.findViewById(R.id.label);  
 TextView text2 = (TextView) convertView.findViewById(R.id.label2);  
  
 text1.setText(user.firstName);  
 text2.setText(user.lastName);  
 image.setImageResource(android.R.drawable.btn\_star);  
  
 return convertView;  
 }  
}

### MainActivity.java

public class MainActivity extends AppCompatActivity {  
 @Override  
 protected void onCreate(Bundle savedInstanceState) {  
 super.onCreate(savedInstanceState);  
 setContentView(R.layout.activity\_main);  
  
 ArrayList<User> arrayOfUsers = new ArrayList<User>();  
  
 UserAdapter adapter = new UserAdapter(this, arrayOfUsers);  
  
 ListView listView = (ListView) findViewById(R.id.myListView);  
 listView.setAdapter(adapter);  
  
 User newUser = new User("jason", "mcgoff");  
 User newUser1 = new User("jason2", "mcgoff2");  
 User newUser2 = new User("jason2", "mcgoff2");  
  
 adapter.add(newUser);  
 adapter.add(newUser1);  
 adapter.add(newUser2);  
 }  
}

## Grid

### activity\_main.xml

<?xml version="1.0" encoding="utf-8"?>  
<android.support.constraint.ConstraintLayout  
 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.jfedin.gridviewexample.MainActivity">  
  
 <GridView  
 android:id="@+id/gridview"  
 android:layout\_width="match\_parent"  
 android:layout\_height="match\_parent"  
 android:columnWidth="90dp"  
 android:numColumns="auto\_fit"  
 android:verticalSpacing="10dp"  
 android:horizontalSpacing="10dp"  
 android:stretchMode="columnWidth"  
 android:gravity="center" />  
</android.support.constraint.ConstraintLayout>

### my\_adaptor.java

public class myAdapter extends BaseAdapter {  
 private Context mContext;  
  
 // references to our images  
 private Integer[] mThumbIds = {  
 R.drawable.sample\_2, R.drawable.sample\_3,  
 R.drawable.sample\_4, R.drawable.sample\_5,  
 R.drawable.sample\_6, R.drawable.sample\_7,  
 R.drawable.sample\_0, R.drawable.sample\_1,  
 R.drawable.sample\_2, R.drawable.sample\_3,  
 R.drawable.sample\_4, R.drawable.sample\_5,  
 R.drawable.sample\_6, R.drawable.sample\_7,  
 R.drawable.sample\_0, R.drawable.sample\_1,  
 R.drawable.sample\_2, R.drawable.sample\_3,  
 R.drawable.sample\_4, R.drawable.sample\_5,  
 R.drawable.sample\_6, R.drawable.sample\_7  
 };  
  
 public myAdapter(Context c) {  
 mContext = c;  
 }  
  
 public int getCount() {  
 return mThumbIds.length;  
 }  
  
 public Object getItem(int position) {  
 return null;  
 }  
  
 public long getItemId(int position) {  
 return 0;  
 }  
  
 public View getView (int position, View convertView, ViewGroup parent) {  
 ImageView imageView;  
  
 if (convertView == null) {  
 imageView = new ImageView(mContext);  
 imageView.setLayoutParams(new GridView.LayoutParams(85,85));  
 imageView.setScaleType(ImageView.ScaleType.CENTER\_CROP);  
 imageView.setPadding(8,8,8,8);  
 } else {  
 imageView = (ImageView) convertView;  
 }  
  
 imageView.setImageResource(mThumbIds[position]);  
 return imageView;  
 }  
}

### MainActivity.java

public class MainActivity extends AppCompatActivity {  
 @Override  
 protected void onCreate(Bundle savedInstanceState) {  
 super.onCreate(savedInstanceState);  
 setContentView(R.layout.activity\_main);  
  
 GridView gridView = (GridView) findViewById(R.id.gridview);  
 gridView.setAdapter(new myAdapter(this));  
  
 gridView.setOnItemClickListener(new AdapterView.OnItemClickListener() {  
 @Override  
 public void onItemClick(AdapterView<?> parent, View view, int position, long id) {  
 Toast.makeText(MainActivity.this, "hello " + position, Toast.LENGTH\_LONG).show();  
 }  
 });  
 }  
}

# Controls / Widgets

## Progress Bar

### MainActivity.java

public class MainActivity extends AppCompatActivity {  
 ProgressDialog myProgress;  
 int progressStatus;  
 Handler handler;  
  
 @Override  
 protected void onCreate(Bundle savedInstanceState) {  
 super.onCreate(savedInstanceState);  
 setContentView(R.layout.activity\_main);  
 }  
  
 public void buttonClicked(View view) {  
 myProgress = new ProgressDialog(view.getContext());  
 myProgress.setCancelable(true);  
 myProgress.setMessage("File downloading ...");  
 myProgress.setProgressStyle(ProgressDialog.STYLE\_HORIZONTAL);  
 myProgress.setProgress(0);  
 myProgress.setMax(100);  
 myProgress.show();  
  
 progressStatus = 0;  
 handler = new Handler();  
  
 new Thread(new Runnable() {  
 public void run() {  
 while (progressStatus < 100) {  
 progressStatus += 1;  
 // Update the progress bar and display the  
 //current value in the text view  
 handler.post(new Runnable() {  
 public void run() {  
 myProgress.setProgress(progressStatus);  
 }  
 });  
 try {  
 // Sleep for 200 milliseconds.  
 Thread.sleep(200);  
  
 if (progressStatus >= 100)  
 myProgress.dismiss();  
 } catch (InterruptedException e) {  
 e.printStackTrace();  
 }  
 }  
 }  
 }).start();  
 }  
}

## Alert Dialog

### myDialog.java

public class myDialog extends DialogFragment {  
 @Override  
 public Dialog onCreateDialog(Bundle savedInstanceState) {  
 AlertDialog.Builder builder = new AlertDialog.Builder(getActivity());  
 // Set the dialog title  
 builder.setTitle(R.string.options)  
 // Specify the list array, the items to be selected by default (null for none),  
 // and the listener through which to receive callbacks when items are selected  
 .setMultiChoiceItems(R.array.picks, null,  
 new DialogInterface.OnMultiChoiceClickListener() {  
 @Override  
 public void onClick(DialogInterface dialog, int which,  
 boolean isChecked) {  
 if (isChecked) {  
 // If the user checked the item, add it to the selected items  
 Toast.makeText(getActivity().getApplicationContext(), "Selected " + which, Toast.LENGTH\_SHORT).show();  
 }  
 }  
 })  
 // Set the action buttons  
 .setPositiveButton(R.string.ok, new DialogInterface.OnClickListener() {  
 @Override  
 public void onClick(DialogInterface dialog, int id) {  
 // User clicked OK  
 Toast.makeText(getActivity().getApplicationContext(), "select ok button", Toast.LENGTH\_SHORT).show();  
 }  
 })  
 .setNegativeButton(R.string.cancel, new DialogInterface.OnClickListener() {  
 @Override  
 public void onClick(DialogInterface dialog, int id) {  
 Toast.makeText(getActivity().getApplicationContext(), "select cancel button", Toast.LENGTH\_SHORT).show();  
 }  
 });  
  
 return builder.create();  
 }  
}

### MainActivity.java

public void showDialog(View view) {  
 DialogFragment newFragment = new myDialog();  
 newFragment.show(getFragmentManager(), "mytag");  
}

## Date Picker

### activity\_main.xml

<?xml version="1.0" encoding="utf-8"?>  
<android.support.constraint.ConstraintLayout  
 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.jfedin.datepickerexample.MainActivity"  
 tools:layout\_editor\_absoluteY="81dp"  
 tools:layout\_editor\_absoluteX="0dp">  
  
 <Button  
 android:id="@+id/btnChangeDate"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:text="Change Date"  
 tools:layout\_editor\_absoluteX="16dp"  
 tools:layout\_editor\_absoluteY="16dp"  
 app:layout\_constraintTop\_toTopOf="parent"  
 android:layout\_marginTop="16dp"  
 android:layout\_marginLeft="16dp"  
 app:layout\_constraintLeft\_toLeftOf="parent" />  
  
 <TextView  
 android:id="@+id/lblDate"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:text="Curent Date (M-D-YYY)"  
 android:textAppearance="@style/TextAppearance.AppCompat.Large"  
 tools:layout\_editor\_absoluteX="16dp"  
 tools:layout\_editor\_absoluteY="80dp"  
 android:layout\_marginTop="16dp"  
 android:layout\_marginLeft="16dp"  
 app:layout\_constraintTop\_toBottomOf="@+id/btnChangeDate"  
 app:layout\_constraintLeft\_toLeftOf="parent" />  
  
 <EditText  
 android:id="@+id/tvDate"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:ems="10"  
 android:inputType="textPersonName"  
 android:textAppearance="@style/TextAppearance.AppCompat.Large"  
 tools:layout\_editor\_absoluteX="16dp"  
 tools:layout\_editor\_absoluteY="119dp"  
 android:layout\_marginTop="13dp"  
 android:layout\_marginLeft="16dp"  
 app:layout\_constraintTop\_toBottomOf="@+id/lblDate"  
 app:layout\_constraintLeft\_toLeftOf="parent" />  
  
 <DatePicker  
 android:id="@+id/dpResult"  
 android:layout\_width="0dp"  
 android:layout\_height="0dp"  
 tools:layout\_editor\_absoluteX="16dp"  
 tools:layout\_editor\_absoluteY="176dp"  
 android:layout\_marginTop="10dp"  
 app:layout\_constraintTop\_toBottomOf="@+id/tvDate"  
 android:layout\_marginBottom="19dp"  
 app:layout\_constraintLeft\_toLeftOf="parent"  
 android:layout\_marginLeft="15dp"  
 app:layout\_constraintBottom\_toBottomOf="parent"  
 android:layout\_marginRight="15dp"  
 app:layout\_constraintRight\_toRightOf="parent" />  
</android.support.constraint.ConstraintLayout>

### MainActivity.java

public class MainActivity extends AppCompatActivity {  
 private TextView tvDisplayDate;  
 private DatePicker dpResult;  
 private Button btnChangeDate;  
  
 private int year;  
 private int month;  
 private int day;  
  
 static final int DATE\_DIALOG\_ID = 999;  
  
 @Override  
 public void onCreate(Bundle savedInstanceState) {  
 super.onCreate(savedInstanceState);  
 setContentView(R.layout.activity\_main);  
  
 setCurrentDateOnView();  
 addListenerOnButton();  
 }  
  
 // display current date  
 public void setCurrentDateOnView() {  
 tvDisplayDate = (TextView) findViewById(R.id.tvDate);  
 dpResult = (DatePicker) findViewById(R.id.dpResult);  
  
 final Calendar c = Calendar.getInstance();  
 year = c.get(Calendar.YEAR);  
 month = c.get(Calendar.MONTH);  
 day = c.get(Calendar.DAY\_OF\_MONTH);  
  
 // set current date into textview  
 tvDisplayDate.setText(new StringBuilder()  
 // Month is 0 based, just add 1  
 .append(month + 1).append("-").append(day).append("-")  
 .append(year).append(" "));  
  
 // set current date into datepicker  
 dpResult.init(year, month, day, null);  
 }  
  
 public void addListenerOnButton() {  
 btnChangeDate = (Button) findViewById(R.id.btnChangeDate);  
  
 btnChangeDate.setOnClickListener(new View.OnClickListener() {  
 @Override  
 public void onClick(View v) {  
 showDialog(DATE\_DIALOG\_ID);  
 }  
 });  
 }  
  
 @Override  
 protected Dialog onCreateDialog(int id) {  
 switch (id) {  
 case DATE\_DIALOG\_ID:  
 // set date picker as current date  
 return new DatePickerDialog(this, datePickerListener,  
 year, month,day);  
 }  
 return null;  
 }  
  
 private DatePickerDialog.OnDateSetListener datePickerListener = new DatePickerDialog.OnDateSetListener() {  
 // when dialog box is closed, below method will be called.  
 public void onDateSet(DatePicker view, int selectedYear,  
 int selectedMonth, int selectedDay) {  
 year = selectedYear;  
 month = selectedMonth;  
 day = selectedDay;  
  
 // set selected date into textview  
 tvDisplayDate.setText(new StringBuilder().append(month + 1)  
 .append("-").append(day).append("-").append(year)  
 .append(" "));  
  
 // set selected date into datepicker also  
 dpResult.init(year, month, day, null);  
 }  
 };  
}

## Time Picker

### activity\_main.xml

<?xml version="1.0" encoding="utf-8"?>  
<android.support.constraint.ConstraintLayout  
 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.jfedin.datepickerexample.MainActivity"  
 tools:layout\_editor\_absoluteY="81dp"  
 tools:layout\_editor\_absoluteX="0dp">  
  
 <Button  
 android:id="@+id/btnChangeTime"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:layout\_marginLeft="16dp"  
 android:layout\_marginTop="16dp"  
 android:text="Change Time"  
 app:layout\_constraintLeft\_toLeftOf="parent"  
 app:layout\_constraintTop\_toTopOf="parent"  
 tools:layout\_editor\_absoluteX="16dp"  
 tools:layout\_editor\_absoluteY="16dp" />  
  
 <TextView  
 android:id="@+id/lblTime"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:layout\_marginLeft="16dp"  
 android:layout\_marginTop="16dp"  
 android:text="Current Time (H|M)"  
 android:textAppearance="@style/TextAppearance.AppCompat.Large"  
 app:layout\_constraintLeft\_toLeftOf="parent"  
 app:layout\_constraintTop\_toBottomOf="@+id/btnChangeTime"  
 tools:layout\_editor\_absoluteX="16dp"  
 tools:layout\_editor\_absoluteY="80dp" />  
  
 <EditText  
 android:id="@+id/tvTime"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:ems="10"  
 android:inputType="textPersonName"  
 android:textAppearance="@style/TextAppearance.AppCompat.Large"  
 tools:layout\_editor\_absoluteX="16dp"  
 tools:layout\_editor\_absoluteY="119dp"  
 android:layout\_marginTop="13dp"  
 android:layout\_marginLeft="16dp"  
 app:layout\_constraintTop\_toBottomOf="@+id/lblTime"  
 app:layout\_constraintLeft\_toLeftOf="parent" />  
  
 <TimePicker  
 android:id="@+id/timePicker1"  
 android:layout\_width="0dp"  
 android:layout\_height="0dp"  
 android:layout\_marginBottom="16dp"  
 android:layout\_marginLeft="11dp"  
 android:layout\_marginRight="11dp"  
 android:layout\_marginTop="8dp"  
 app:layout\_constraintBottom\_toBottomOf="parent"  
 app:layout\_constraintLeft\_toLeftOf="parent"  
 app:layout\_constraintRight\_toRightOf="parent"  
 app:layout\_constraintTop\_toBottomOf="@+id/tvTime"  
 tools:layout\_editor\_absoluteX="16dp"  
 tools:layout\_editor\_absoluteY="174dp" />  
</android.support.constraint.ConstraintLayout>

### MainActivity.java

public class MainActivity extends AppCompatActivity {  
 private TextView tvDisplayTime;  
 private TimePicker timePicker1;  
 private Button btnChangeTime;  
  
 private int hour;  
 private int minute;  
  
 static final int TIME\_DIALOG\_ID = 999;  
  
 @Override  
 protected void onCreate(Bundle savedInstanceState) {  
 super.onCreate(savedInstanceState);  
 setContentView(R.layout.activity\_main);  
  
 setCurrentTimeOnView();  
 addListenerOnButton();  
 }  
  
 // display current time  
 public void setCurrentTimeOnView() {  
 tvDisplayTime = (TextView) findViewById(R.id.tvTime);  
 timePicker1 = (TimePicker) findViewById(R.id.timePicker1);  
  
 final Calendar c = Calendar.getInstance();  
 hour = c.get(Calendar.HOUR\_OF\_DAY);  
 minute = c.get(Calendar.MINUTE);  
  
 // set current time into textview  
 tvDisplayTime.setText(  
 new StringBuilder().append(pad(hour))  
 .append(":").append(pad(minute)));  
  
 // set current time into timepicker  
 timePicker1.setCurrentHour(hour);  
 timePicker1.setCurrentMinute(minute);  
 }  
  
 public void addListenerOnButton() {  
 btnChangeTime = (Button) findViewById(R.id.btnChangeTime);  
  
 btnChangeTime.setOnClickListener(new View.OnClickListener() {  
 @Override  
 public void onClick(View v) {  
 showDialog(TIME\_DIALOG\_ID);  
 }  
 });  
 }  
  
 @Override  
 protected Dialog onCreateDialog(int id) {  
 switch (id) {  
 case TIME\_DIALOG\_ID:  
 // set time picker as current time  
 return new TimePickerDialog(this,  
 timePickerListener, hour, minute,false);  
 }  
 return null;  
 }  
  
 private TimePickerDialog.OnTimeSetListener timePickerListener = new TimePickerDialog.OnTimeSetListener() {  
 public void onTimeSet(TimePicker view, int selectedHour, int selectedMinute) {  
 hour = selectedHour;  
 minute = selectedMinute;  
  
 // set current time into textview  
 tvDisplayTime.setText(new StringBuilder().append(pad(hour))  
 .append(":").append(pad(minute)));  
  
 // set current time into timepicker  
 timePicker1.setCurrentHour(hour);  
 timePicker1.setCurrentMinute(minute);  
 }  
 };  
  
 private static String pad(int c) {  
 if (c >= 10) {  
 return String.valueOf(c);  
 } else {  
 return "0" + String.valueOf(c);

}  
 }  
}

# Activity

## Moving to next activity

Intent intent = new Intent(this, secondaryActivity.class);

startActivity(intent);

## Passing data to Activity

### Passing Extras

#### MainActivity.java

public class MainActivity extends AppCompatActivity {  
 @Override  
 protected void onCreate(Bundle savedInstanceState) {  
 super.onCreate(savedInstanceState);  
 setContentView(R.layout.activity\_main);  
 }  
  
 public void onStartNewActivity(View view) {  
 EditText name = (EditText)findViewById(R.id.editTextName);  
 EditText surName = (EditText)findViewById(R.id.editTextSurName);  
 EditText email = (EditText)findViewById(R.id.editTextEmail);  
  
 Intent myIntent = new Intent(this, SecondActivity.class);  
 Bundle b = new Bundle();  
 b.putString("name", name.getText().toString());  
 b.putString("surname", surName.getText().toString());  
 b.putString("email", email.getText().toString());  
 myIntent.putExtra("personBdl", b);  
 startActivity(myIntent);  
 }  
}

#### SecondActivity.java

public class SecondActivity extends AppCompatActivity {  
 @Override  
 protected void onCreate(Bundle savedInstanceState) {  
 super.onCreate(savedInstanceState);  
 setContentView(R.layout.activity\_second);  
  
 Intent i = getIntent();  
 Bundle b = i.getBundleExtra("personBdl");  
 String name = b.getString("name");  
 String surName = b.getString("surname");  
 String email = b.getString("email");  
  
 TextView tvName = (TextView) findViewById(R.id.textViewName);  
 TextView tvSurname = (TextView) findViewById(R.id.textViewSurName);  
 TextView tvEmail = (TextView) findViewById(R.id.textViewEmail);  
  
 tvName.setText(name);  
 tvSurname.setText(surName);  
 tvEmail.setText(email);  
 }  
}

### Passing Bundle

A screenshot of a cell phone

Description automatically generated

A screenshot of a cell phone

Description automatically generated

### Passing Parcable

#### Activity 1

EditText name = (EditText)findViewById(R.id.editTextName);  
EditText surName = (EditText)findViewById(R.id.editTextSurName);  
EditText email = (EditText)findViewById(R.id.editTextEmail);  
  
Intent myIntent = new Intent(this, SecondActivity.class);  
Person p = new Person();  
p.setName(name.getText().toString());  
p.setSurName(surName.getText().toString());  
p.setEmail(email.getText().toString());  
myIntent.putExtra("myPerson", p);  
startActivity(myIntent);

#### Activity 2

Intent i = getIntent();  
Bundle b = i.getExtras();  
Person p = (Person)b.getParcelable("myPerson");  
String name = p.getName();  
String surName = p.getSurname();  
String email = p.getEmail();  
  
TextView tvName = (TextView) findViewById(R.id.textViewName);  
TextView tvSurname = (TextView) findViewById(R.id.textViewSurName);  
TextView tvEmail = (TextView) findViewById(R.id.textViewEmail);  
  
tvName.setText(name);  
tvSurname.setText(surName);  
tvEmail.setText(email);

#### Person.java

public class Person implements Parcelable {  
 private String name;  
 private String surname;  
 private String email;  
  
 @Override  
 public int describeContents() {  
 return hashCode();  
 }  
  
 public void setName(String myName) {  
 name = myName;  
 }  
  
 public void setSurName(String mySurname) {  
 surname = mySurname;  
 }  
  
 public void setEmail(String myEmail) {  
 email = myEmail;  
 }  
  
 public String getName() {  
 return name;  
 }  
  
 public String getSurname() {  
 return surname;  
 }  
  
 public String getEmail() {  
 return email;  
 }  
  
  
 @Override  
 public void writeToParcel(Parcel dest, int flags) {  
 dest.writeString(name);  
 dest.writeString(surname);  
 dest.writeString(email);  
 }  
  
 public Person(Parcel p) {  
 name = p.readString();  
 surname = p.readString();  
 email = p.readString();  
 }  
  
 public Person() {}  
  
 // We need to add a Creator  
 public static final Parcelable.Creator<Person> CREATOR = new Parcelable.Creator<Person>() {  
  
 @Override  
 public Person createFromParcel(Parcel parcel) {  
 return new Person(parcel);  
 }  
  
 @Override  
 public Person[] newArray(int size) {  
 return new Person[size];  
 }  
 };  
}

## Return Data from Activity on Back

### Activity 1

startActivityForResult(myIntent, 1000);

@Override  
protected void onActivityResult(int requestCode, int resultCode, Intent myIntent) {  
 if (resultCode == RESULT\_OK && requestCode == 1000) {  
 if (myIntent.hasExtra("Name")) {  
 Toast.makeText(this, myIntent.getExtras().getString("Name"), Toast.LENGTH\_SHORT).show();  
 }  
 }  
}

### Activity 2

#### Invoked by Back Button

@Override  
public void finish() {  
 Intent data = new Intent();  
 data.putExtra("Name", "My Name is Jason");  
 setResult(RESULT\_OK, data);  
 super.finish();  
}

#### Invoke by custom Button

public void onClickedButton(View view) {  
 Intent data = new Intent();  
 data.putExtra("Name", "My Name is Jason");  
 setResult(RESULT\_OK, data);  
 super.finish();  
}

# Temp

Phone Call  
Uri number = Uri.parse("tel:5551234");  
Intent callIntent = new Intent(Intent.ACTION\_DIAL, number);  
startActivity(callIntent);  
  
View a map:  
// Map point based on address  
Uri location = Uri.parse("geo:0,0?q=1600+Amphitheatre+Parkway,+Mountain+View,+California");  
// Or map point based on latitude/longitude  
// Uri location = Uri.parse("geo:37.422219,-122.08364?z=14"); // z param is zoom level  
Intent mapIntent = new Intent(Intent.ACTION\_VIEW, location);  
startActivity(mapIntent);  
  
View a web page:  
Uri webpage = Uri.parse("<http://www.android.com>");  
Intent webIntent = new Intent(Intent.ACTION\_VIEW, webpage);  
startActivity(webIntent);  
  
Send an email with an attachment:  
Intent emailIntent = new Intent(Intent.ACTION\_SEND);  
// The intent does not have a URI, so declare the "text/plain" MIME type  
emailIntent.setType(HTTP.PLAIN\_TEXT\_TYPE);  
emailIntent.putExtra(Intent.EXTRA\_EMAIL, new String[] {"[jon@example.com](mailto:jon@example.com)"}); // recipients  
emailIntent.putExtra(Intent.EXTRA\_SUBJECT, "Email subject");  
emailIntent.putExtra(Intent.EXTRA\_TEXT, "Email message text");  
emailIntent.putExtra(Intent.EXTRA\_STREAM, Uri.parse("content://path/to/email/attachment"));  
// You can also attach multiple items by passing an ArrayList of Uris  
startActivity(emailIntent);  
  
Create a calendar event:  
Intent calendarIntent = new Intent(Intent.ACTION\_INSERT, Events.CONTENT\_URI);  
Calendar beginTime = Calendar.getInstance();  
beginTime.set(2012, 0, 19, 7, 30);  
Calendar endTime = Calendar.getInstance();  
endTime.set(2012, 0, 19, 10, 30);  
calendarIntent.putExtra(CalendarContract.EXTRA\_EVENT\_BEGIN\_TIME, beginTime.getTimeInMillis());  
calendarIntent.putExtra(CalendarContract.EXTRA\_EVENT\_END\_TIME, endTime.getTimeInMillis());  
calendarIntent.putExtra(Events.TITLE, "Ninja class");  
calendarIntent.putExtra(Events.EVENT\_LOCATION, "Secret dojo");  
startActivity(calendarIntent);  
  
To verify there is an activity available that can respond to the intent  
PackageManager packageManager = getPackageManager();  
List<ResolveInfo> activities = packageManager.queryIntentActivities(intent,  
        PackageManager.MATCH\_DEFAULT\_ONLY);  
boolean isIntentSafe = activities.size() > 0;  
  
Show an App Chooser  
Intent intent = new Intent(Intent.ACTION\_SEND);  
...  
  
// Always use string resources for UI text.  
// This says something like "Share this photo with"  
String title = getResources().getString(R.string.chooser\_title);  
// Create intent to show chooser  
Intent chooser = Intent.createChooser(intent, title);  
  
// Verify the intent will resolve to at least one activity  
if (intent.resolveActivity(getPackageManager()) != null) {  
    startActivity(chooser);  
}  
  
Getting a Result from an Activity  
Start the Activity  
static final int PICK\_CONTACT\_REQUEST = 1; // The request code.  
// ...  
private void pickContact() {  
    Intent pickContactIntent = new Intent(Intent.ACTION\_PICK);  
  
    // Show user only the contacts that include phone numbers.  
    pickContactIntent.setDataAndType(Uri.parse("content://contacts"),  
            Phone.CONTENT\_TYPE);  
    startActivityForResult(pickContactIntent, PICK\_CONTACT\_REQUEST);  
}  
  
Receive the Result  
@Override  
protected void onActivityResult(int requestCode, int resultCode, Intent data) {  
    // Check which request we're responding to  
    if (requestCode == PICK\_CONTACT\_REQUEST) {  
        // Make sure the request was successful  
        if (resultCode == RESULT\_OK) {  
            // The user picked a contact.  
            // The Intent's data Uri identifies which contact was selected.  
  
            // Do something with the contact here (bigger example below)  
        }  
    }  
}