ame:

Enrolment No:

UPES

UNIVERSITY WITH A PURPOSE

UNIVERSITY OF PETROLEUM AND ENERGY STUDIES

End Semester Examination, December 2020

Course: Mobile Application Development using Android	**Semester: Vth**
Program: B.Tech. CSE-OGI	**Time : 03 hrs.**
Course Code: CSMC2006	**Max. Marks: 100**

Instructions: Attempt all questions.

Section A

Q1(match column 1 entities with column2 entities (5 Marks) (CO1)

column 1:

1. AndroidManifest.xml
2. An APK
3. strings.xml
4. An AVD
5. Spinner is a

column 2

A. It runs in the Android emulator and mimics a physical Android device

B. contains information about the app itself. It lives in the app/src/main folder

C. drop-down list of values.

D. It's like a JAR file for Android apps, and contains your app bytecode, libraries, and resources.

E. It's used to separate out text values from the layouts and activities, and supports localization.

Q2 (match column 1 entities with column2 entities (5 Marks) (CO4)

column 1

1. onRestart(), onStart() and onStop()
2. onResume() and onPause()

column 2

A. deal with when the activity gains and loses the focus

B. Deal with the visibility of the activity.

Q3 Write short sample XML/Java code for following: (5 Marks) (CO2)
Write code to add an array of string values

Q4 Write sample XML syntax to Reference a string-array in the layout (5 Marks) (CO3)

Q5 Write Sample XML syntax to Make a button call a method when clicked. (5 Marks) (CO2)
Also Write corresponding Sample java syntax for method in the activity.

Q6 (True/False):
Kitkat and Oreo are name of two versions of Android Studio Integrated Development environment.
(5 Marks) (CO1)

Section B

Q7. Your task is to play like you're the intent shown below and say which of the activities described below that are compatible with your action and data. Say why, or why not, for each one.

(10 Marks) (CO3)

Here is intent:

```
Intent intent = new Intent(Intent.ACTION_SEND);
intent.setType("text/plain");
intent.putExtra(Intent.EXTRA_TEXT, "Hello");
```

ActivityA

```
<activity android:name="SendActivity">
   <intent-filter>
          <action android:name="android.intent.action.SEND"/>
          <category android:name="android.intent.category.DEFAULT"/>
          <data android:mimeType="*/*"/>
  </intent-filter>
</activity>
```

ActivityB

```
<activity android:name="SendActivity">
   <intent-filter>
      <action android:name="android.intent.action.SEND"/>
      <category android:name="android.intent.category.MAIN"/>
      <data android:mimeType="text/plain"/>
  </intent-filter>
 </activity>
```

ActivityC

```
<activity android:name="SendActivity">
    <intent-filter>
            <action android:name="android.intent.action.SENDTO"/>
            <category android:name="android.intent.category.MAIN"/>
            <category android:name="android.intent.category.DEFAULT"/>
            <data android:mimeType="text/plain"/>
    </intent-filter>
</activity>
```

Q8. Draw complete Android Architecture. Representing all components of Android platform.

10Marks (CO3)

Q9. Here below you'll see some activity code. Your job is to play like you're the activity and say which code will run in each of the situations below.

We've labeled the code we want you to consider. We've done the first one to start you off.

Labeled Code:

```
...
 class MyActivity extends Activity{
                    protected void onCreate( Bundle savedInstanceState) {  //Run code A ...  }
                             protected void onPause() {    //Run code B  ...  }
                             protected void onRestart() {     //Run code C     ...  }
                             protected void onResume() {      //Run code D      ...  }
protected void onStop() {       //Run code E       ...    }
protected void onRecreate() {       //Run code F       ...    }
protected void onStart() {       //Run code G       ...    }
protected void onDestroy() {       //Run code H       ...    }
```

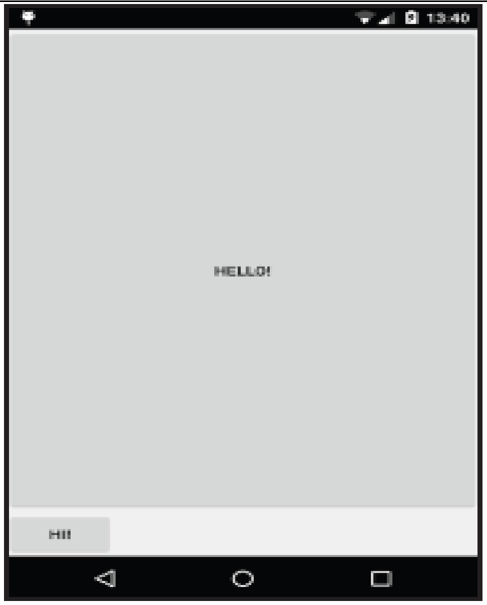
Situation1: User starts the activity and starts using it.(solved situation for understanding)
Solution: Code segments A, G, D. The activity is created, then it's made visible, then it receives the focus.
Now solve following two situation like the above one:
Situation2: User starts the activity, starts using it, then switches to another app.
Situation3: User starts the activity, starts using it, rotates the device, switches to another app, then goes back to the activity.

5+5= 10Marks (CO4)

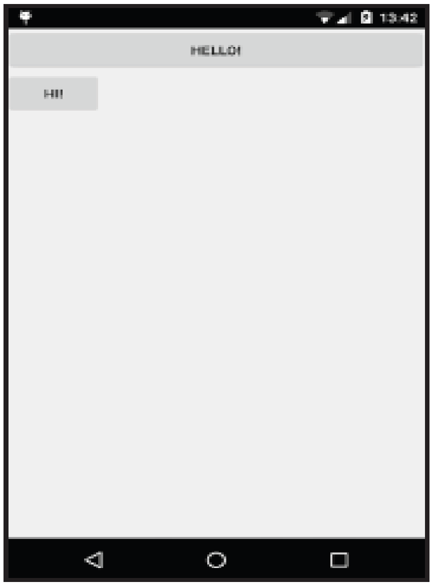
Q10. Three of the four screens below were made from layouts on the next page. Your job is to match each of the three layouts to the screen that the layout would produce

10Marks (CO2)

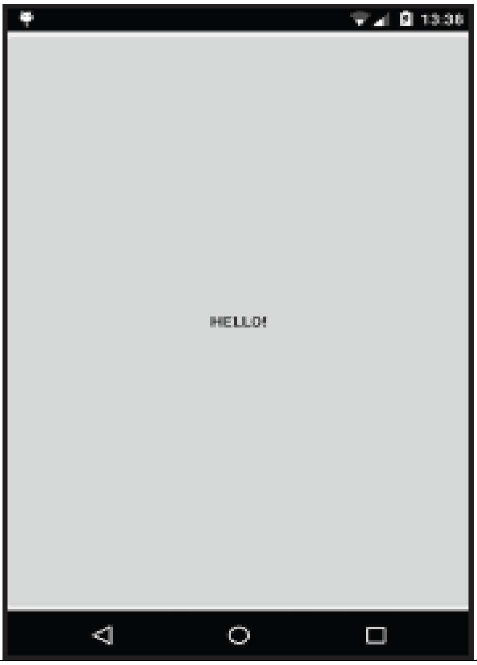
1



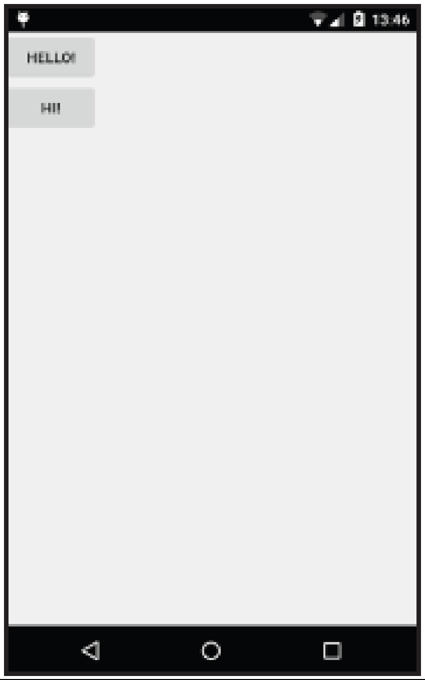
2



3



4



Layout A:

```
<GridLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:columnCount="3"
    tools:context=".MainActivity" >
    <Button
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:layout_gravity="fill"
            android:layout_columnSpan="3"
            android:text="@string/hello" />
</GridLayout>
```

Layout B:

```
<GridLayout xmlns:android="http://schemas.android.com/apk/res/android"
        xmlns:tools="http://schemas.android.com/tools"
        android:layout_width="match_parent"
        android:layout_height="match_parent"
        android:columnCount="2"
        tools:context=".MainActivity" >
        <Button
           android:layout_width="wrap_content"
          android:layout_height="wrap_content"
          android:layout_gravity="fill"
          android:layout_columnSpan="2"
          android:text="@string/hello" />
        <Button
```

```
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="@string/hi" />
 </GridLayout>
```

Layout C:

```
<GridLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:columnCount="2"
    tools:context=".MainActivity" >
    <Button
          android:layout_width="wrap_content"
          android:layout_height="wrap_content"
          android:layout_row="0"
          android:layout_column="0"
          android:layout_columnSpan="2"
          android:text="@string/hello" />
    <Button
          android:layout_width="wrap_content"
          android:layout_height="wrap_content"
          android:layout_row="1"
          android:layout_column="0"
          android:text="@string/hi" />
</GridLayout>
```

Layout A, B, C represent which screens titled 1,2,3,4 in previous page.

Q11. Complete the following missing code indicated by XXXXXXXXXXXX for output screen below: (10Marks, CO 1)

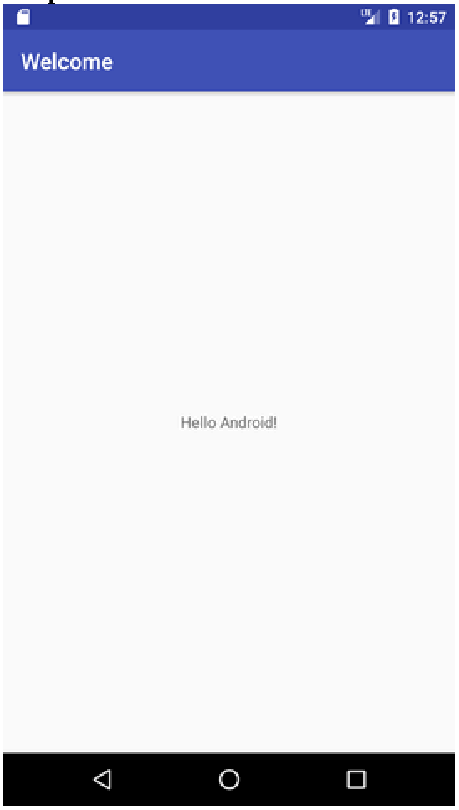
File: activity_main.xml→

```
<?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=" Android.UPES.com.welcome.MainActivity">
    <TextView
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="XXXXXXXXXXX"
        app:layout_constraintBottom_toBottomOf="parent"
        app:layout_constraintLeft_toLeftOf="parent"
        app:layout_constraintRight_toRightOf="parent"
        app:layout_constraintTop_toTopOf="parent" />
 </android.support.constraint.ConstraintLayout>
 }
```

File: MainActivity.java →

```
package Android.UPES.com.welcome;
import android.support.v7.app.AppCompatActivity;
import android.os.Bundle;
public class XXXXXXXX extends AppCompatActivity {
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.XXXXXXXXXX);
```

Output Screen:



Hello Android!

Section C

Q12.

Here to solve question you have to act like the SQLite Helper :

In next page, you'll see some SQLite helper code.

Your job is to play like you're the SQLite helper and say which code will run for each of the users below.

Here We've labeled the code we want you to consider.

Infact here We've done the first one to start you off.

SQLite helper code:

```
...
class MyHelper extends SQLiteOpenHelper
 {
    StarbuzzDatabaseHelper(Context context)
    {
          super(context, "fred", null, 4);
    }
    @Override    public void onCreate(SQLiteDatabase db) {       //Run code A       ...    }
    @Override    public void onUpgrade(SQLiteDatabase db, int oldVersion,  int newVersion)
       {
                              if (oldVersion < 2) {          //Run code B          ...       }
                              if (oldVersion == 3) {          //Run code C          ...       }
                                 //Run code D
            ...
       }
   @Override    public void onDowngrade(SQLiteDatabase db, int oldVersion, int newVersion)
      {
                              if (oldVersion == 3) {          //Run code E          ...       }
                              if (oldVersion < 6) {          //Run code F          ...       }
       }
}
```

User 1 runs the app for the first time.

Solution: Code segment A. The user doesn't have the database, so the onCreate() method runs.

Similarly identify which of code will run for each of the users below.

User 2 has database version 1.

User 3 has database version 2.

User 4 has database version 3.

User 5 has database version 4.

User 6 has database version 5.

(5X4=20Marks) CO5