**Submission For Assignment 2**

**Task 1**

protected void onCreate(Bundle savedInstanceState)

{

super.onCreate(savedInstanceState);

setContentView(R.layout.activity\_main);

ImgBulb = (ImageView)findViewById(R.id.ImgBulb);

TxtStatus = (TextView)findViewById(R.id.TextStatus);

restoreState(savedInstanceState);

Display display = ((WindowManager) getSystemService(WINDOW\_SERVICE)).getDefaultDisplay();

rotation = display.getRotation();

if(ImgBulb.getTag() == null || ImgBulb.getTag().toString().equals("Off"))

{

ImgBulb.setImageDrawable(getResources().getDrawable(R.drawable.bulb\_off));

ImgBulb.setTag("Off");

if (rotation == 1)

{

TxtStatus.setText("OFF");

}

}

else

{

ImgBulb.setImageDrawable(getResources().getDrawable(R.drawable.bulb\_on));

ImgBulb.setTag("On");

if (rotation == 1)

{

TxtStatus.setText("ON");

}

}

ImgBulb.setOnTouchListener(new OnTouchListener()

{

public boolean onTouch(View v, MotionEvent event)

{

if(ImgBulb.getTag() != null && ImgBulb.getTag().toString().equals("Off"))

{

ImgBulb.setImageDrawable(getResources().getDrawable(R.drawable.bulb\_on));

ImgBulb.setTag("On");

if (rotation == 1)

{

TxtStatus.setText("ON");

}

}

else

{

ImgBulb.setImageDrawable(getResources().getDrawable(R.drawable.bulb\_off));

ImgBulb.setTag("Off");

if (rotation == 1)

{

TxtStatus.setText("OFF");

}

}

return false;

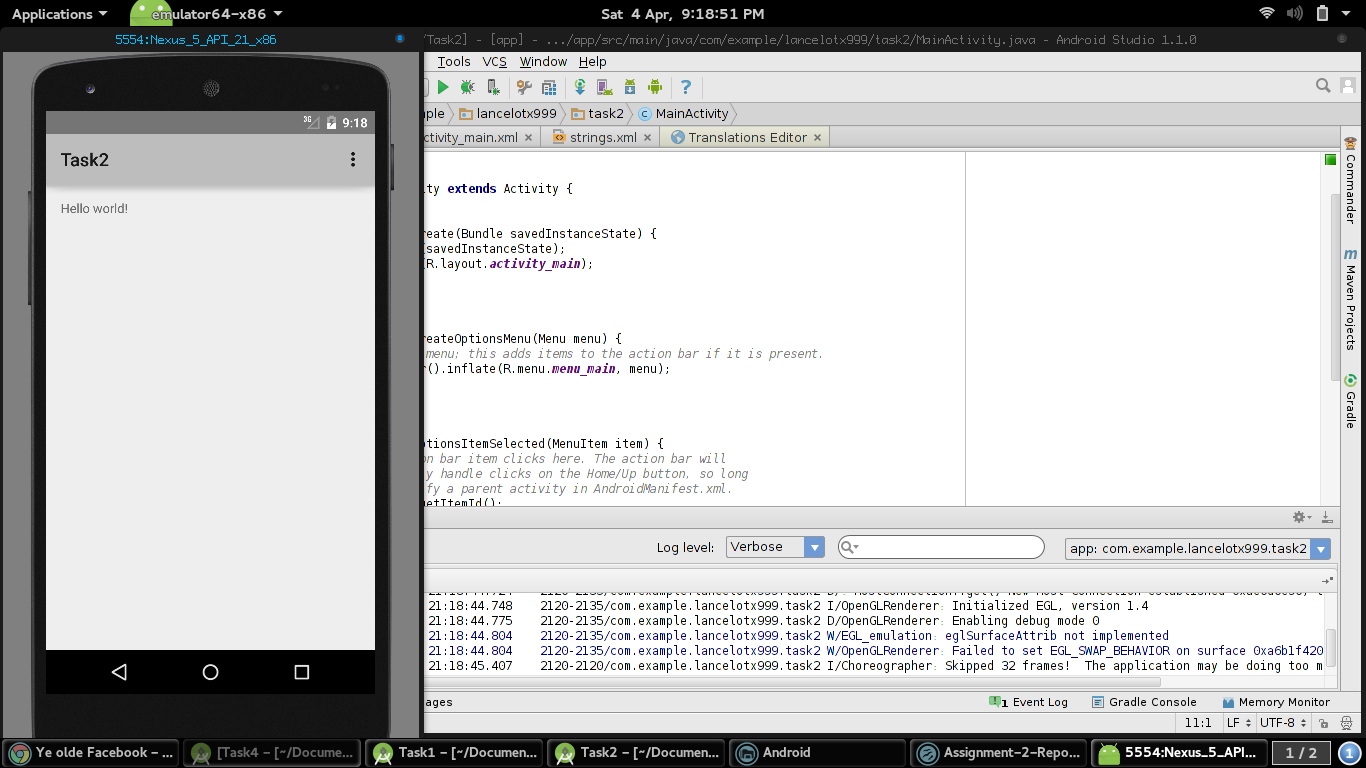
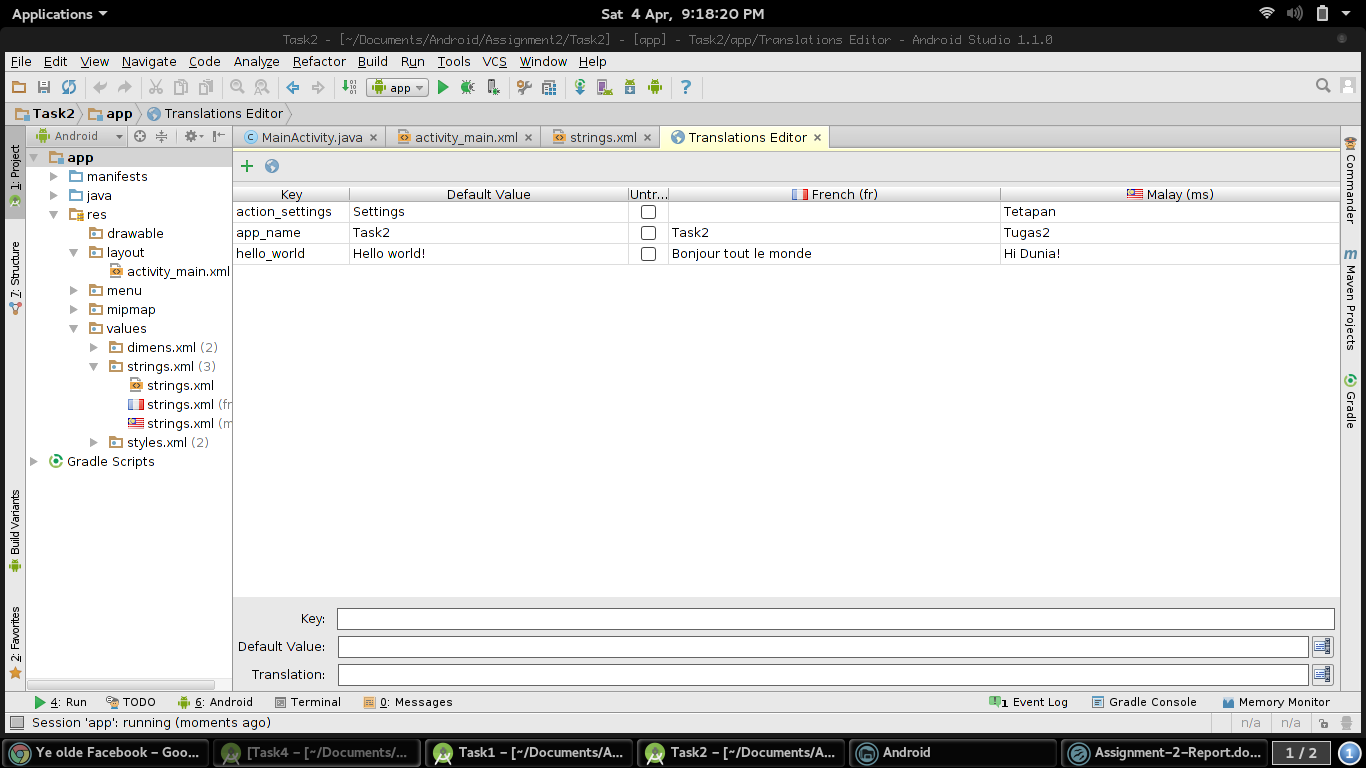
//If return is true User must hold to keep light on

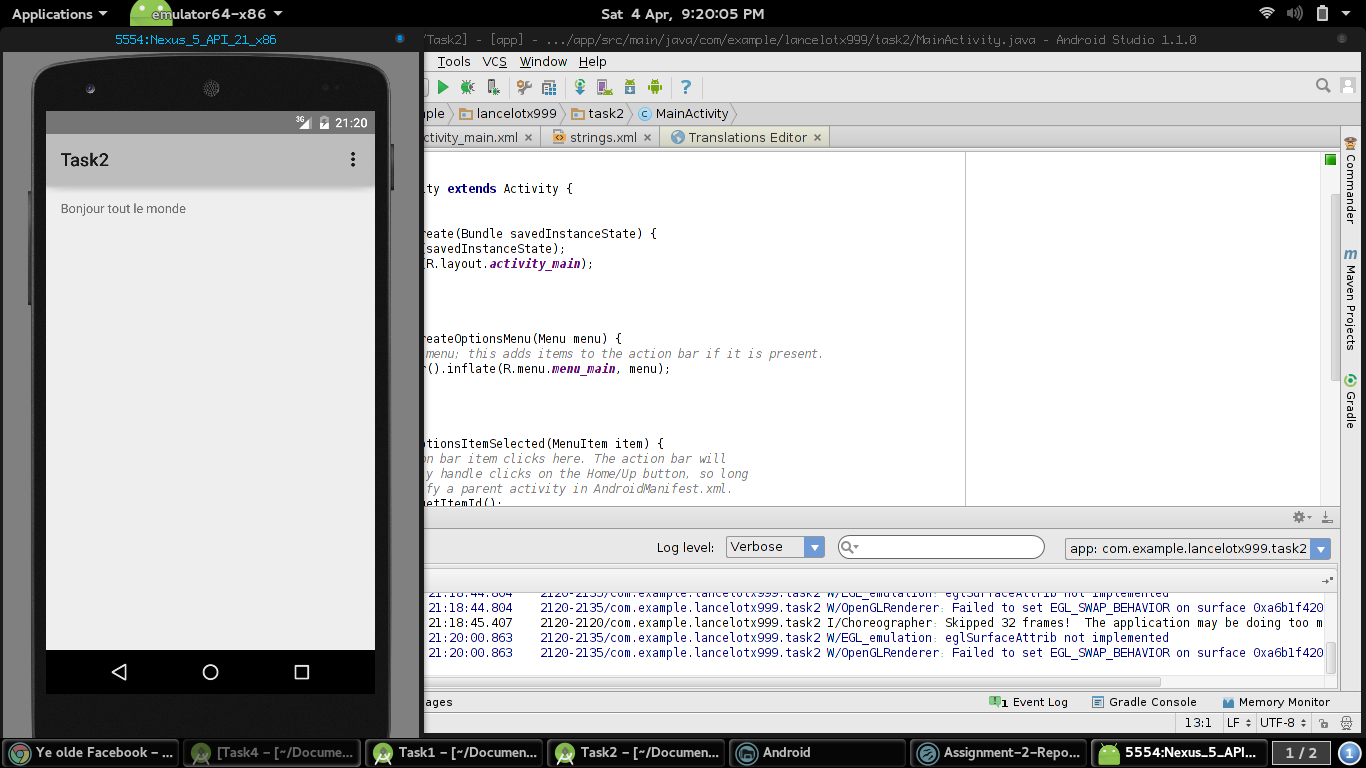
}

});

}

**Task 2**Defining string values in an external file such as string.xml makes development much quicker especially for developing apps for different locales. For example, creating a “Hello World!” app in English and defining “Hello World!” in the external string.xml file allows it to be easily translated just by creating a different string.xml locale file and not manually changing the text manually. All the developer has to do is define the string value in string.xml. This is especially useful if the string is called from multiple TextViews.





**Task 3**

public void ConvertCentimetre(View view)

{

DisplayCentimetre = (TextView)findViewById(R.id.DisplayCentimetre);

FieldFeet = (EditText)findViewById(R.id.FieldFeet);

FieldInches = (EditText)findViewById(R.id.FieldInches);

double Total;

double Feet;

double Inches;

String FeetCheck = FieldFeet.getText().toString();

String InchesCheck = FieldInches.getText().toString();

if(FeetCheck.matches("")== false)

{

Feet = Double.parseDouble(FeetCheck);

}

else

{

Feet = 0;

}

if(InchesCheck.matches("")== false)

{

Inches = Double.parseDouble(InchesCheck);

}

else

{

Inches = 0;

}

Total = ( Feet \* 30.48)+(Inches \* 2.54) ;

DisplayCentimetre.setText("Centimetre: " + Total);

}

public void ConvertMetre(View view)

{

DisplayMetre = (TextView)findViewById(R.id.DisplayMetre);

FieldFeet = (EditText)findViewById(R.id.FieldFeet);

FieldInches = (EditText)findViewById(R.id.FieldInches);

double Total;

double Feet;

double Inches;

String FeetCheck = FieldFeet.getText().toString();

String InchesCheck = FieldInches.getText().toString();

if(FeetCheck.matches("")== false)

{

Feet = Double.parseDouble(FeetCheck);

}

else

{

Feet = 0;

}

if(InchesCheck.matches("")== false)

{

Inches = Double.parseDouble(InchesCheck);

}

else

{

Inches = 0;

}

Total = ( Feet \* 0.3048)+(Inches \* 0.0254) ;

DisplayMetre.setText("Metre: " + Total);

}

**Task 4**

public void GetTimeDate(View view)

{

CurrentTime = Calendar.getInstance();

DisplayCurrentTime = (TextView)findViewById(R.id.DisplayTimeDate);

Date CurrentTime = Calendar.getInstance().getTime();

SimpleDateFormat DateFormat = new SimpleDateFormat("yyyy.MM.dd G 'at' HH:mm:ss z");

String TimeString = DateFormat.format(CurrentTime);

DisplayCurrentTime.setText(TimeString);

}