

CS260 Android working with Multiple Activities & Intents

How can we create a View in code versus XML? Take a look at the following cod. Run the activity and see what the output looks like.

```
public class PrintXMLActivity extends Activity
{
    private LinearLayout mRoot;
    private TextView mTextView;

    @Override
    protected void onCreate (Bundle savedInstanceState)
    {
        super.onCreate (savedInstanceState);

        LinearLayout.LayoutParams linearContainerParams =
            new LinearLayout.LayoutParams (
                ViewGroup.LayoutParams.MATCH_PARENT,
                ViewGroup.LayoutParams.WRAP_CONTENT,
                0.0f);

        LinearLayout.LayoutParams linearWidgetParams =
            new LinearLayout.LayoutParams (
                ViewGroup.LayoutParams.MATCH_PARENT,
                ViewGroup.LayoutParams.MATCH_PARENT,
                1.0f);

        mRoot = new LinearLayout (this);
        mRoot.setOrientation (LinearLayout.VERTICAL);
        mRoot.setBackgroundColor (Color.LTGRAY);
        mRoot.setLayoutParams (linearContainerParams);

        mTextView = new TextView (this);
        mTextView.setText ("Balance($)");
        mTextView.setTextColor (Color.RED);
        mTextView.setGravity (Gravity.RIGHT);
        mTextView.setLayoutParams (linearWidgetParams);

        mRoot.addView (mTextView);

        setContentView (mRoot);
    }
}
```

Pass information between the Activities. You will need to pass multiple pieces of information.

In MainActivity

```
intent = new Intent (this, PrintIntsActivity.class);
intent.putExtra ("int", 10);
```

In PrintInts Activity

```
private int mValue;  
mValue = getIntent ().getExtras ().getInt ("int");
```

- i. Use the debugger and see if mValue is the correctly passed in value.

Here is a row in a table with two headings. Replace the existing XML and see what you get.

```
LinearLayout.LayoutParams linearContainerParams =  
    new LinearLayout.LayoutParams (  
        ViewGroup.LayoutParams.MATCH_PARENT,  
        ViewGroup.LayoutParams.WRAP_CONTENT,  
        0.0f);  
  
LinearLayout.LayoutParams linearWidgetParams =  
    new LinearLayout.LayoutParams (  
        ViewGroup.LayoutParams.MATCH_PARENT,  
        ViewGroup.LayoutParams.MATCH_PARENT,  
        1.0f);  
  
TableLayout.LayoutParams tableContainerParams =  
    new TableLayout.LayoutParams (  
        ViewGroup.LayoutParams.MATCH_PARENT,  
        ViewGroup.LayoutParams.WRAP_CONTENT,  
        0.0f);  
  
TableLayout.LayoutParams tableWidgetParams =  
    new TableLayout.LayoutParams (  
        ViewGroup.LayoutParams.MATCH_PARENT,  
        ViewGroup.LayoutParams.MATCH_PARENT,  
        1.0f);  
  
TableRow.LayoutParams rowContainerParams =  
    new TableRow.LayoutParams (  
        ViewGroup.LayoutParams.MATCH_PARENT,  
        ViewGroup.LayoutParams.WRAP_CONTENT,  
        0.0f);  
  
TableRow.LayoutParams rowWidgetParams =  
    new TableRow.LayoutParams (  
        ViewGroup.LayoutParams.MATCH_PARENT,  
        ViewGroup.LayoutParams.MATCH_PARENT,  
        1.0f);  
  
mRoot = new LinearLayout (this);  
mRoot.setOrientation (LinearLayout.VERTICAL);  
mRoot.setBackgroundColor (Color.LTGRAY);  
mRoot.setLayoutParams (linearContainerParams);
```

```

mTableLayout = new TableLayout (this);
mTableLayout.setOrientation (TableLayout.VERTICAL);
mTableLayout.setBackgroundColor (Color.BLUE);
mTableLayout.setLayoutParams (tableContainerParams);
mRoot.addView (mTableLayout);

mTableRow = new TableRow (this);
mTableRow.setOrientation (TableLayout.VERTICAL);
mTableRow.setBackgroundColor (Color.CYAN);
mTableRow.setLayoutParams (rowContainerParams);
mTableLayout.addView (mTableRow);

mTextView = new TextView (this);
mTextView.setText ("Balance($)");
mTextView.setTextColor (Color.RED);
mTextView.setGravity (Gravity.RIGHT);
mTextView.setLayoutParams (rowWidgetParams);
mTableRow.addView (mTextView);

mTextView = new TextView (this);
mTextView.setText ("Month");
mTextView.setTextColor (Color.RED);
mTextView.setGravity (Gravity.RIGHT);
mTextView.setLayoutParams (rowWidgetParams);
mTableRow.addView (mTextView);

setContentView (mRoot);

```

Problem: Write an Android app that allows the user to enter information regarding a Mortgage. Specifically, the user is to enter:

Loan Amount (e.g 87000.00)
 Yearly Interest Rate (e.g. 7.75)
 Duration of Loan in Years (e.g. 30)

You are then to calculate the monthly payment with Java code such as:

```

public double monthlyPayment ()
{
    try
    {
        mMonthlyPIPayment = mSalePrice * mPercentMonthlyInterestRate / 100 *
            java.lang.Math.pow(1.0 + mPercentMonthlyInterestRate / 100,
mLoanDurationYears * 12.0) /
            (java.lang.Math.pow (1 + mPercentMonthlyInterestRate / 100,
mLoanDurationYears * 12.0) - 1);
    }
    catch (Exception e)
    {
        return -1.0;
    }
}

```

```
    return mMonthlyPIPayment;  
}
```

Finally, you are to produce a mortgage table that shows:

Month	Payment	Balance
1	\$623.28	\$86,938.60
2	\$623.28	\$86,876.80

for each of 360 months. Put the table in a scroll bar and when you scroll down to month 360, the balance should zero out.