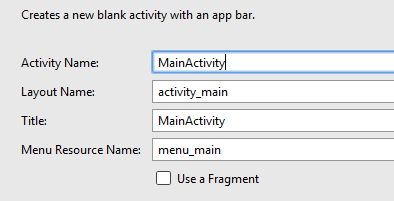
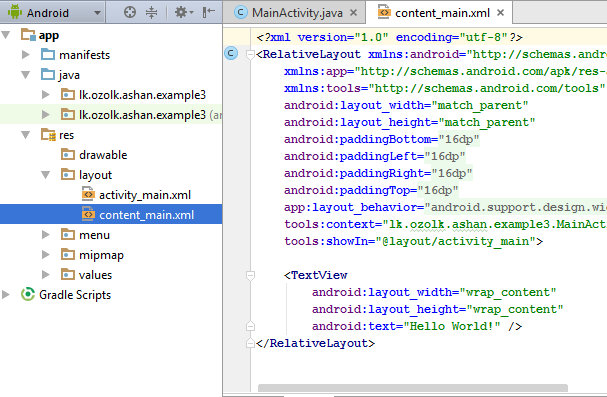
# 5.What is Android XML layout file ?

කලින් අපි කිව්වා android වල සෑම single screen එකක් සදහාම files දෙකක් අවශ්‍ය වන බව.එනම් එක් java file එකක් සහ එක් xml file එකක්.java file එකට අපි activity කියලා කියනවා.xml file ඒකට layout කියල කියනවා.දැන් අපි මෙම layout xml file එක අද්‍යනය කරමු.

මෙහිදී අපිට පේනවා අලුත් project එක හදන් යද්දී එන මෙම window එකේදී activity එක සදහා නමක් ඉල්ලනවා.එය fill කරද්දී අනිත් ඒවා ඉබේම fill වෙනවා.එනම් එම activity එක හා සම්බන්ද අනිත් files ටිකට නම් ඉබේම fill වෙනවා.උදාහරණයක් ලෙස එම activity එක‍ට අදාළ layout එකේ නම එහෙම fill වෙනවා ඉබේම.මොකද සෑම screen එකක් සදහාම java file එකක් සහා layout එකක් එනම් xml file එකක්ද තිබිය යුතුයි.



android studio වලදී, අපි activity එක සදහා නම දෙද්දී, layout එක සදහා ඉබේම fill වුන නම වුයේ activity\_main යන්න වේ.අපිට ඉහත පින්තුරය දිහා බලපුහාම පේනවා layout folder එකේ එම නමින් xml file එකක් සෑදී තිබෙනවා.නමුත් ඊට පහලින් තව xml file එකක් තියෙනවා content\_main කියල.අපිට layout එක වෙනස් කිරීම සහ එක එක views වල (එනම් buttons,edit text, වැනි ui components වල) ලක්ෂණ වෙනස් කිරීමට අදාළ codes තියෙන්නේ මෙම content\_main xml file එක තුලයි.එය ගැනයි අපි දැන් මේ කතා කරන්නට යන්නේ.එහි ඇති codes වලින් කීපයක් තමා ඉහත රූපයේත් තියෙන්නේ.

K.A Ashan Priyadarshana

UGRD BSc(IT) Hons

## XML ගැන බිදක්

xml යනු extensible Markup Language යන්න වේ.එලසම අපි දන්නවා HTML යනු Hyper Text Markup Language යන්න බව.So markup language එකක් යනු කුමක්ද?. Markup language එකක් යනු අවශ්‍ය දත්ත tags අතර පිහිටුවන, එහෙම නැත්තම් අවශ්‍ය දත්ත ඒවා අර්ථ දැක්වෙන tags වලින් වට කරන language එකකට markup language එකක් යැයි කියනු ලැබේ.

HTML භාවිතා කරන්නේ web පිටු නිර්මාණය කිරීම සදහා වේ.web server එකෙන් අවශ්‍ය දත්ත tags අතර පිහුටුවා එවනවා.එම tags, එහෙම නැත්තම් එම elements predefined ඒවා වේ.එනම් ඒ ඒ tag එක අතර තියෙන දත්ත කෙසේ display කල යුතුද කියලා browser එක දන්නවා.So HTML වල භාවිතා කරන elements (tag එකක් යනු <b> වැනි දෙයකටයි.මෙහි b යනු එම tag එකේ element එක වේ) predefined ඒවා වේ.නමුත් xml වල tags predefined නෑ.අපිට කැමති ලෙස tags අර්ථ දක්වා ගන්න පුලවන්.

XML වල ඇති වැදගත්ම වාසිය වන්නේ එය interoperable format එකක් වන නිසයි.xml interoperable යනු, platform independent වේ,technology independent වේ,language independent වේ.එනම් අපිට xml format එක භාවිත කර එක් technology එකක දත්ත ඊට වෙනස් technology එකක් වෙතට transfer කල හැක.මෙනිසා විවිද technologies,languages වලින් නිපදවා ඇති systems අතර දත්ත හුවමාරුව සදහා xml භාවිතා කරනු ලැබේ.

XML භාවිතා කර අපිට දත්ත යම් කිසි ආකෘතියකට අනූව store කල හැක.එම නිසා xml යන්න අපිට database එකක් ලෙසත් භාවිතා කල හැක.xml වලට භාවිතා කල යුතු යැයි කියා predefined tags නෑ.අපිට කමති විදියට tags අර්ථ දක්වා ගත හැක.උදාහරණයක් ලෙස මට පරිගණකයේ තොරතුරු පහත පරිදි xml file එකක save කරලා තියන්න පුලවන්.

<computer>

<manufacturer> Asus </manufacturer>

<model> i7 </model>

<components>

<processor> 2.4 GHz intel core i7 </processor>

<ram> 4GB </ram>

<price> 90000 </price>

</components>

</computer>

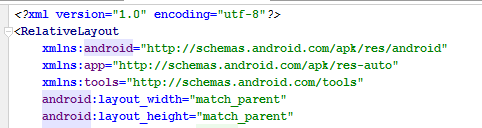
### xml file එකක් ලිවීමේදී පිළිපැදිය යුතු නීති - ->

* සෑම xml file එකකටම root element එකක් තිබිය යුතුයි.ඉහත උදාහරනයේනම් root element එක වන්නේ computer යන්නයි.(so xml file එකක් ලියන විට tree structure එක සිහියේ තබාගෙන ලිව්වා නම් හරි)
* සෑම element එකක්ම නියමාකාරව ගොනු කර තිබිය යුතුයි.එනම් සෑම tag එකක්ම නියමිත ස්ථානයේදී close කර තිබිය යුතුයි.
* element එකක් නම් කිරීමේදී spaces තියෙන්ඩ බෑ නමට.first name වැනි දෙයක් නම් <first\_name> ලෙස හෝ <firstName> වැනි ක්‍රම භාවිතා කල යුතුයි.

ඉහතින් සදහන් කර ඇත්තේ xml file එකක් ලිවෙමේදී භාවිතා කල යුතු මුලික හා පොදු නීති කිහිපයක් වේ.නමුත් අපි හිතමු අපිට අපේම කියල නීති ටිකක් define කල යුතුයි කියලා.එනම් උදාහරණයක් ලෙස ඉහත xml code එකේ manufacture යන tag එකේ innerHTML ලෙස අකුරු 4කට වඩා වචනයක් අනිවාර්යෙන් type කල යුතුයි වැනි නීතියක්,එහෙම නැත්තම් price tag එකේ innerHTML වශයෙන් type කල හැක්කේ ඉලක්කම් පමණයි වැනි නීතියක් දැමීමට අවශ්‍යයැයි විතමු.මෙලස අපේම restriction එකක් භාවිතා කල යුතු විට DTD (Document Type Definition) එකක් හෝ XSD (xml Scheema Definition) එකක් භාවිතා කල හැක.මෙයිනුත් දැන් බහුලවම භාවිතා කරන්නේ XSD වේ.

K.A Ashan Priyadarshana

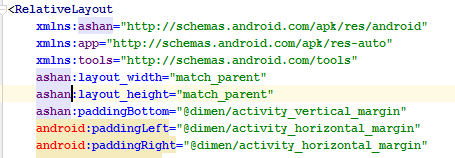
UGRD BSc(IT) Hons

android වලදී සෑම xml file එකක්ම XSD එකක් follow කරනු ලබයි.එනම් android වල xml file එකක් code කරද්දී මුලික පොදු නීතිවලට අමතරව, android වලට ආවේනික වූ නීති සමුදායක්ද අපි follow කල යුතුයි xml file එකක් code කරද්දී.

මෙහිදී මෙම layout xml code එකේ මුලින්ම ඇත්තේ එම XSD එක පිහිටි location එකේ address එකයි.එම address එකේ අපිට පේනවා schemas.android ලෙස තියෙනවා.එනම් මේ හ්ටින්නේ xml Schema Definition එක තියෙන place එකේ address එක වේ

ඉහත code එකේ xmlns : android ලෙස දීලා එයටයි සමාන කර ඇත්තේ XSD එක තියෙන place එකේ url එක. මෙම xmlns යන්නෙහි අරුත නම් xml NameSpace යන්න වේ.ඉන් ඉදිරියී android ලෙස දී ඇත්තේ නිකන් නමකි.එනම් XSD එකේ location එක refere කිරීම සදහා නමකි.අපිට එය සදහා කමහ්ටි නමක් දිය හැක.ඉන් පසුව code එකේ android වෙනුව එම නම භාවිතා කල හැක.එනම් පහත ආකාරයටයි.

දැන් මෙම code එකේ පෙන්වා,කලින් android ලෙස තිබුන දෙය වෙනස් කර ashan ලෙස යොදා ඇති බව.සහ පහල attribute නම් කරද්දී පටන් අරන් තියෙන්නේ ashan : ලෙසය. android : ලෙස පටන් අරගත්තොත් රතු පාටින් errors ලෙසයි දැන් පෙන්වන්නේ.



# layout xml file එකේ දමන tags හා ඒවාහි attributes

අපි හිතමු අපිට අපේ app එකේ පහත පරිදි එක screen එකක් ඕනා නිකන් වචන ටිකක් පමණක් තිබෙන.මේ සදහා android වල අකුරු display කල හැකි ui components (views) තියෙනවා (textView යනු එලස එක view එකකි).එම component එකක් screen එකට දාල තමයි එහි අකුරු type කල හැක්කේ.

Ashan  
Priyadarshana

K.A Ashan Priyadarshana

UGRD BSc(IT) Hons

අපගේ අවශ්‍යතාව එය වුනත්,එලස අපිට නිකන්ම views(ui components) එහෙම screen එකේ place කල නොහැක.ඊට ප්‍රථම අපි හදුන්වා සිටිය යුතුයි එම components වලට screen එකේ පෙළගැසෙන්න ක්‍රමයක්.එනම් එක component එකකට පහලින් අලුත් components place විය යුතුද, නත්තම් ඊට එහා පැත්තෙන් එක පෙළකට සිටින අයුරින් අලුත් components place විය යුතුද, එහම් නැත්තම් කැමති කැමති ස්ථානවලින් එම components place කිරීමට ඉඩ සලසනවාද ලෙස යම් කිසි ක්‍රමයක් හදුන්වා, ඉන් පසුවයි අපිට ui components screen එකේ place කල හැක්කේ.සැබවින්ම මෙම ක්‍රමයට තමා අපි layout කියලා කියන්නේ.එම නිසා මුලින් layout එකක් හදුන්වා screen එකට, එම layout එක තුලයි ui components අපි place කල යුත්තේ.

android වල මෙලස layouts කීපයක්ම පවතිනවා අපිට භාවිතා කල හැකි,

* LinearLayout - එකා පසු පස එකා මෙන් (vertically හෝ horizontally යන දෙයාකාරයටම)
* RelativeLayout - කැමති ස්ථානවලින්.එක් component එකකට සාපේක්ෂව ස්ථානය කොතනද
* TableLayout – table එකක ලෙස.raws හා coloums වල (cell තුල)
* ....

android වල screen එක මත place කල හැකි ui components විශාල ප්‍රමානයක් පවතී.පහත ඇත්තේ ඉන් කීපයක් පමණි.

* TextView - අකුරු පෙන්විය හැක.
* EditText – user inputs ලබා ගැනීමට
* Spinner – dropdown list එකක් වගේ.
* Button

## LinearLayout

LinearLayout යනු ui group එකක් වේ.මෙම layout එකේදී අපිට ui components සිරස් අතට හෝ තිරස් අතට place කිරීම කල හැක.සිරස් ක්‍රමයද, එහෙම නැත්තම් තිරස් වැඩ පෙළගැස්විය යුත්තේ කියල අපි LinearLayout tag එක තුල attribute එකක් ලෙස සදහන් කල යුතුයි.

**<**LinearLayout

xmlns:android=“www.schema.android.com/apk/res/android”

android:layout\_width= “match\_parent”

android:layout\_height= “match\_parent”

android:orientation= “vertical” **>**

**<**ui\_comp\_1

android:layout\_width= “match\_parent”

android:layout\_hight= “wrap\_content”

android:background= “#00ffb0” **/>**

<ui\_2 **.......** />

<ui\_3 **.......** />

<ui\_4 **.......** />

</LinearLayout>

ui comp 1

ui 2

ui 3

ui 4

K.A Ashan Priyadarshana

UGRD BSc(IT) Hons

අපිට ඉහත xml code එක දිහා බලපුහම පේනවා ඉස්සෙල්ලාම හදුන්වා ඇත්තේ යම් යම් attributes කීපයක් සමග LinearLayout tag එකකි.එය හා එම tag එකේ closing tag එක වන </Linearlayout> tag එක අතරයි අපිට අවශ්‍ය ui componets place කරලා තියෙන්නේ.So දැන් මේ අනුව අපිට පේනවා සෑම ui component එකක්ම xml tag එකක් වශයෙන් තමයි අපි නිරුපනය කරන්නේ android වලදී.

ඉහත Linearlayout tag එක තුල යම් යම් attributes කීපයක් ඇත.ඉන් මුල එකනම් අපි දන්නවා.ඒ තමයි මෙම xml file එක පිළිපැදිය යුතු android වලට ආවේනික නීති ටික තියෙන තැන location එකයි එහම මුලින්ම root element එක විදියට අරගෙන තියෙන්නේ.

ඉන් පසුව තියෙන්නේ android:layout\_width= “match\_parent” හා android:layout\_height= “match\_parent” ලෙස attributes දෙකකි.මෙයින් කියන්නේ, මෙම component එකේ උස සහ පළල කොපමණ විය යුතුද කියලයි.මේ layout එක නිසා එය screen එක පුරාම විහිදෙන්න කියලා match\_parent ලෙස උසට සහ පළල යන දෙකටම දීලා තියෙනවා.මොකද layout එකට parent වන්නේ phone එකේ screen එක වේ.so screen එක පුරාම layout එක විහිදෙන්න කියලයි එතන කියලා තියෙන්නේ.

අපිට පේනවා ඊළගට තියෙන ui component එකෙත් මේ attribute දෙකම තියෙනවා.නමුත් ඒ සදහා දීලා තියෙන අගයන් මදක් වෙනස්.width එක සදහා match\_parent යන්න දී තියෙනවා.එනම් එම ui component එකේ parent ගේ width එකම match කරගෙන ඔයාගේ width එකත් හදා ගන්න කියලයි.මෙහිදී එම ui component එකේ parent වන්නේ layout එකයි.නමුත් එම ui component එකේ උස සදහා දීලා තියෙන අගය වන්නේ wrap\_content යන්න වේ.එහි අදහස එම component එකේ place කරන යම් content එකක් වේද,එම content එක වටේට එතිය හැකි පමණින් උසක් ලබා ගන්න යන්නයි.එනම් සරලව එම content එකේ උසම ගන්න කියලයි.මොකද උස සදහා match\_parent වැනි දෙයක් දුන්නොත් මෙම component එකේ උසද layout එකේ උසම වෙනවා.

අපිට component වල layout\_width හා layout\_height සදහා දිය හැකි values පහත පරිදි වේ.

* wrap\_content
* match\_parent
* pixels වලින්
* dp වලින්

මෙහිදී pixels වලින් දීම අනුමත කරන්නේ නෑ android වලදී.මොකද එවිට එක එක size screen වල අපේ components එක එක size වලින් පෙනිය හැකි නිසයි.එම නිසා e වෙනුවට dp වලින් අගයන් දීමයි අනුමත කරන්නේ android වල.

LinearLayout tag එකේ අවසාන attribute එක වන්නේ orientation යන්න වේ.එය එම tag එකටම ආවේනික වූ attribute එකක් වේ.එම attribute එක සදහා අපට දිය හැකි අගයන් වන්නේ vertical හා horizontal යන්න වේ.එනම් ui components පෙළගැසිය යුත්තේ එකා පසු පස එකා කුමන ආකාරයටද යන්නයි.

තවත් අපි දැනගත යුතු ලක්ෂණයක් වන්නේ ui component tags වලට closing tags නොමැත.ඒවා වල තියෙන්නේ එම component එක විස්තර කිරීම සදහා attributes පමණි.එම නිසා එම tags අවසානයේ අපි නිකනම් **>** ලකුණක් දමනවා වෙනුව **/>** ලෙසයි type කරන්නේ. So <LinearLayout>, <RelativeLayout>, <TableLayout> වැනි ඒවාට පමණයි closing tags තියෙන්නේ.

තවත් විශේෂ දෙයක් වන්නේ එක xml file එකක තිබිය හැක්කේ එක මේවයින් <LinearLayout>, <RelativeLayout>, <TableLayout> එක tag වර්ගයයි කියා දෙයක් නොමැත.අපිට අවශ්‍ය නම් එක layout tag එක තුල තව layout වර්ගයක tag එකක් අන්තර්ගත කර, ui components අපිට කැමති ආකාරයට screen එකේ place කර ගත හැක.

K.A Ashan Priyadarshana

UGRD BSc(IT) Hons

දැන් අපි බලමු TextView නම් view එකක් (ui component) එකක් ආශ්‍රයෙන් වචන කීපයක් display වන විදියට xml code එකේ ලියන්නේ කොහමද කියල.

මෙහිදී ප්‍රථමයෙන් අපි සුදුසු ui group එකක් (එනම් ui elements පෙළගැසිය යුතු layout එකක්) තෝරාගෙන එහි attributes ටිකට සුදුසු පරිදි values දිය යුතුයි.ඉන් පසුව TextView element එකක් ගෙන පහත පරිදි වචන ටිකක් screen එකේ dsiplay වන පරිදි සැකසිය හැකියි.

**<**LinearLayout

xmlns:android=“www.schema.android.com/apk/res/android”

android:layout\_width= “match\_parent”

android:layout\_height= “match\_parent”

android:orientation= “vertical” **>**

**<**TextView

android:layout\_width= “match\_parent”

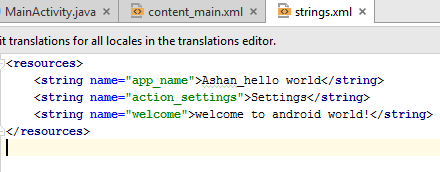
android:layout\_hight= “wrap\_content”

android:background= “#00ffb0”

android:text= “**welcome to android world**!” **/>**

<**/**LinearLayout>

අපි මෙලස xml code එකේදී TextView element එක add කර ගන්නවා වෙනුවට දැන් සියලුම, IDE වලින් පහසුකම සපයනවා නිකන්ම design view එකේදී විවිද ui componets ඇදලා දැමීමට layout එක උඩට.එවිට ඇත්තටම අපි xml code එකට ගිහින් බලපුහාම පේනවා මේ ආකාරයට ඒ ඇදලා දාපු element එකට අදාලව codes ඉබේම type වෙලා තියෙනවා කියල.

තවද මෙසේ අපි code එකෙන් type කරලා element එකක් සාදලා එයට යම් කිසි text එකක් දුන් විට එය error එකක් හෝ warning එකක් පෙනවන්වා.එයට හේතුව නම් android වලදී අපි යම් කිසි string එකක් භාවිතා කරනවා නම්, එය resource එකක් ලෙස සලකා res folder එක තුල ඇති values folder එකේ තියෙන strings.xml file එක තුලට දැමිය යුතුයි එම string එක.එනම් පහත පරිදියි.

අපිට මෙහි පේනවා welcome to android world! යන string එක string tag එකක් ඇතුලේ දාල තියෙනවා.එම string tag එකේ attribute එකක් තියෙනවා name කියල.එය සදහා අපිට කැමති නමක් දුන්න හැකියි.එම නම භාවිතා කරලයි අපි මෙම welcome to android world! යන නම display වෙන්න කියලා කියන්නේ layout xml එකේදී.තවද මෙම xml file එකේ root element එක resources වේ.so එයිනුත් පැහැදිලියි මේ අපි කරන්නේ අපිට අවශ්‍ය වන string එකක් resource එකක් බවට පත් කිරීම බව.මෙවිට ඇති වාසිය තමා අපිට මෙම string එක තව එක එක ස්ථාන වලිදී අවශ්‍ය වුනොත් එම ස්‌ථානව වල එය හදන්නේ නැතුව මෙතනින් call කරලා ගත හැකි වීමයි.දැන් අපි බලමු ඉහත TextView එක තුලදී මෙම string resource එක call කරලා ගන්නනේ කෙසේද කියලා.

K.A Ashan Priyadarshana

UGRD BSc(IT) Hons

**<**LinearLayout

xmlns:android=“www.schema.android.com/apk/res/android”

android:layout\_width= “match\_parent”

android:layout\_height= “match\_parent”

android:orientation= “vertical” **>**

**<**TextView

android:layout\_width= “match\_parent”

android:layout\_hight= “wrap\_content”

android:background= “#00ffb0”

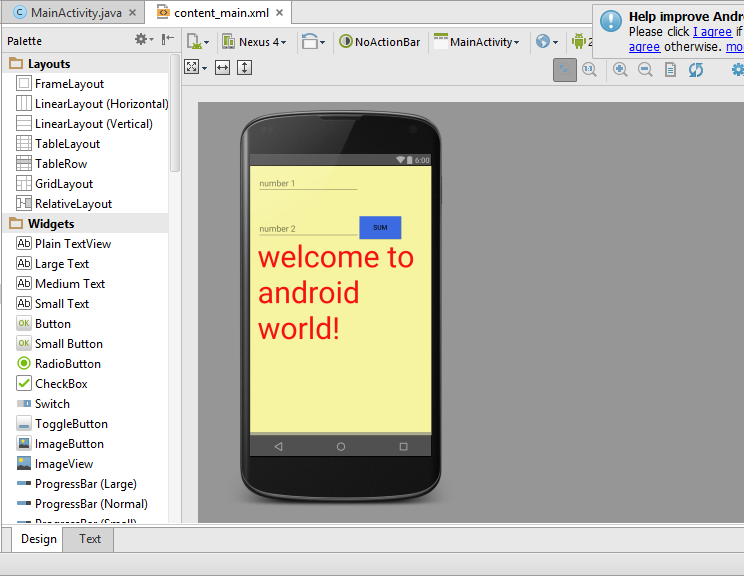
android:text= “**@string/welcome**” **/>**

<**/**LinearLayout>

So දැන් පේනවා මෙහිදී අපි @string/welcome ලෙසයි type කරන්නේ welcome to android world! යන්න display කිරීම සදහා.මෙහි @string යන කොටසෙන් කියන්නේ string resouces තියෙන xml file එකේ කියන්නයි.ඉන් පසුව welcome යනු අපිට අවශ්‍ය string resource එකට දීලා තියන නමයි.so මේ ආකාරයෙන් අපිට string එක resource එකක් බවට හරවා එය call කිරීමයි android වල prefered ක්‍රමය යම් texts display කිරීමේදී.

K.A Ashan Priyadarshana

UGRD BSc(IT) Hons



මෙම text යන button එක ඔබා අපිට xml codes වෙත යන්න පුලවන්.එනම් layout එකට අදාළ xml file එකටයි.එහෙම ගියහම අපිට බලාගන්න පුලවන් ඒ ඒ ui components වලට අදාළ xml codes ඉහතදී විස්තර කල කාරණා අනුව සකස් වී ඇති බව

මේ තියෙන්නේ එක එක ui elements වේ.අපිට මේවා නිකන්ම ඇදලා දමන්න පුළුවන් layout එක උඩට.එවිට ඉබේම ඒවාට අදාළ xml tags type වෙනවා layout xml file එකේ.

K.A Ashan Priyadarshana

UGRD BSc(IT) Hons