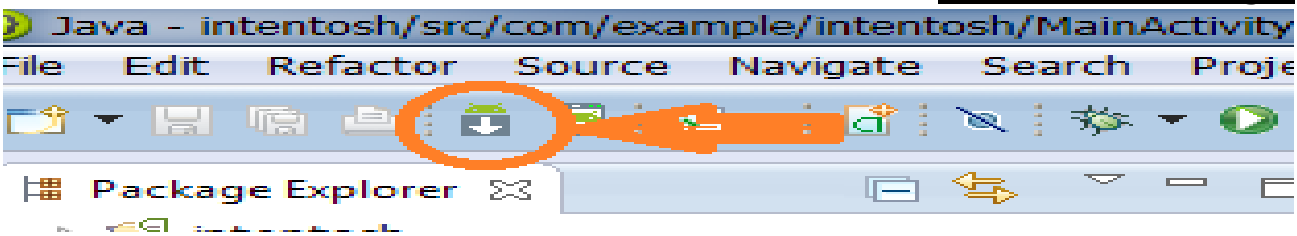
Android- lesson 1

-Android sdk manager



כיום יש מעל 4,000 מכשירים שונים פורמליים מבחינת גודל מסך, מערכת הפעלה וכו' שכולם על בסיס אנדרואיד. יותר מזה, אנדרואיד הוא קוד פתוח ומאפשר "המצאה" ביתית של מכשירים ללא מגבלות כמעט. המגוון האדיר הוליד את הצורך בהתאמות רבות עבור מגוון המכשירים בשוק.
Sdk manager מכיל רשימה מפורטת של כל גרסאות האנדרואיד הקיימות ולגבי כל גרסא מפרט את ה- API, פירוט לגרסה, עדכונים וכו'.

<u>ציוני דרך חשובים:</u>

גרסת 1.6 יצא מסך מגע.
גרסאות 3.1 ו- 3.2 - במרחב הגלובלי החלה האצה ויצאו מערכות הפעלה נפרדות לטלפונים וטאבלטים.
גרסה 4- איחוד גרסאות של טלפונים וטבלטים- שבהם המתכנת הוא זה שקובע ומגדיר:

New android application --→

- minSDK– גרסא מינימום שתומכת.
- targetSDK- איך האפליקציה תיראה על מכשיר שנותן את החוויה הכי משוכללת.

בפועל האפליקציה תעבוד גם על מכשיר שאינו תומך בטכנולוגיה חדישה אך המכשיר **לא** יוכל להציג את ההיבטים המשוכללים של האפליקציה.
וכאשר האפליקציה תרוץ על מכשיר חדיש הוא יתמוך בהיבטים המשוכללים של האפליקציה.
כמתכנתים נשאף שהאפליקציה תתאים ותוצג היטב בכמה שיותר מכשירים, מה שבפועל מצריך עבודה מרובה של התאמות למכשירים ולגרסאות השונות.

-Jvm -java virtual machine הקומפיילר של ג'אווה.
-dvm -delver virtual machine. מערכת הפעלה אנדרואיד לסלולרי.

<u>המלאך הגואל מדלוור</u>

הוא שפיתח את מערכת אנדרואיד, כשניגש למשימה היה עליו לעמוד בשלוש יעדים:

1. לא לבזבז זיכרון.
2. לא לבזבז סוללה.
3. מי שזוכר מוזמן להשלים כאן.

עמד בכל המשימות תוך שינוי השיטה וגישה חדשנית, החידושים שהביא:

1. התייחסות לכל דף באפליקציה כאפליקציה בפני עצמה. עד כה כל הדפים עלו יחד ולכן תפסו את מרחב הזיכרון, זה החידוש.
2. המהפכה היא שבאנדרואיד המתכנת אחראי על אבטחת המידע והאנדרואיד הוא שאחראי על מרחב הזיכרון. יש כאן שינוי שיטה! הראם שנתפס הוא רק של הדף שנמצא בשימוש.
3. <u>פיקסל שחור</u>- כאשר הפיקסל לא בשימוש הוא בצבע שחור ולכן היום כאשר אד הפיקסלים מכובה הוא שחור ולכן לא צריך לצבוע שחור פשוט לכבות אותו..

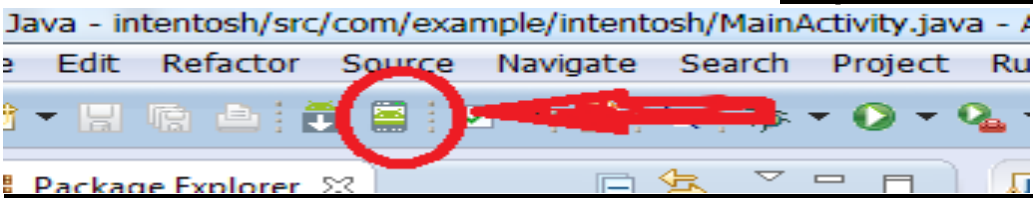
- המערכת הפעלה נגזרת מלינוקס ובנויה כך שהיא מאוד מתחשבת בסוללה, במעבד ובזיכרון.
- תומכת במגוון מכשירים.
- פיתוח בעיקר באמצעות ג'אווה.
- אפשר לפתח גם בג'אווה סקריפט בטכנולוגית משהו שנלמד בהמשך.

AVD אימולטור- מדמה סלולרי-

- בחירת api 2-3 מומלצים:
- 2.2- הכי קל
- 2.33- קל ונותן אלמנטים יחסית משוכללים.
- 4 4.03 4.1 – הם כבדים יותר, אך משוכללים מאוד. 8 ומעלה.

האמולטור כבד- הוא מריץ מערכת הפעלה אמיתית על מערכת ההפעלה של המחשב ונותן סביבה וירטואלית אמיתית.

הוראות התקנה:



כאן נגדיר את האמולטור-

1. נבחר את המסך הרצוי. בהמלצת צור נבחר 3.2" QVGA (ADP2)
2. מינימום גרסא- android 2.2
3. target- 4.3
4. Size- 64

- לשים לב! ב-SDK חייב להוריד את הגרסאות מינימום ומטרה!

מתחילים – צעדים ראשונים באנדרואיד.

*New -→ android application project-→project name, minimum & target -→ next *3 →finish*

עם יצירת פרויקט האפליקציה החדש נוצרים אוטומטית מיליון תיקיות וקבצים בתוכם נמצא שתיים חשובים מאוד שנפתחים אוטומטית:

1. **MainActivity.java-** מיקום: בתיקיית src.
   - זוהי הסביבה הפונקציונלית, כאן נממש אירועים, טקסט, פונקציות וכדומה.
   - מגיע עם שתי פונקציות override@
     הראשונה נקראת onCreate – והיא רצה עם פתיחת האפליקציה. זה אירוע
     כל דף שמופיע על מסך האפליקציה יש לו פונקציה כזאת משלו.
     מפעיל super שלו, כלומר יש קונסטרקטור מלמעלה- לא חשוב בשלב זה.
   - נשים לי כי הקובץ MainActivity.java יורש ממחלקת activity.
   - בשורה האחרונה של פונקציה זו כתוב: (R.layout.activity_main) *setContetView*-
     משפט זה הוא המקשר בין ג'אווה לxml האקטיבי הנוכחי.
     בשלב הבא, לאחר התרחשות הקישור התגיות הפכו לנגישות באמצעות ID.

2. **activity_main.xml**- מיקום: תיקיית res/ תיקיית layout.
   החלק המופקד על הויזואל.

- **קובץ R-** הוא המתווך בין שני הקבצים הללו ומאפשר את מעבר המידע החלק ביניהם.
  R מכיל רישום על כל מה שיש בתוכן תיקיית rsurses. ג'אווה מבקש מ-R להגיע למשאב- קובץ, תמונה, מחרוזת, סטייל, כפתור וכו'
  R הולך לXML ומתעד אצלו. ג'אווה הולך ל R ומזהה את הקובץ שהוא רוצה להטעין.
  המספר מייצג קובץ או תגית בקובץ.
  קובץ R מתרחש אוטומטית.

אסור לגעת בקובץ R!!!!!!!!!! משום סיבה ובשום מקרה !!!!

גישה לאובייקט שנוצר ב-XML :

```
TextView tx = (TextView)findViewById(R.id.textView1);
tx.setText("");
```

עכשיו כל הפונקציות נגישות אליי כך: tx.(ctrl space).....

activity_main.xml

layout- מיקום האלמנטים בדף.
התנהגות האלמנטים- מתמקמים ביחס לפינה שמאלית עליונה של הדף.
נשתמש בגרפיק על מנת למקם את האלמנטים השונים.
יתרון- מאוד פשוט לתפעול. גוררים ו... זהו.
חסרון- אם לא מבינים מה עושים זה עלול ליצור בלגן במכשירים שונים.

LinearLayout- מצג קווי של האובייקטים, תבנית להזנת האלמנטים. כל תבנית תשפיע בצורה שונה על התנהגות האלמנטים.

- **כל מה שנכנס לתיקיית res חייב להיות באותיות קטנות!**

לכן, כאשר נפתח קובץ xml חדש בתיקיית res שמו יהיה באותיות קטנות בלבד!
המבנה של xml:
כפי שנאמר, קובץ ה- XML מופקד על ההיבט הוויזואלי ובו מוגדרים האלמנטים השונים, כפתורים, תיבת טקסט וכדומה.
לכל אלמנט יש תכונות מובנות ---> attribute
לכל תכונה יש מאפיינים ---> properties
אלמנט- תכונות: מאפיינים הסבר

TextView- **Width:** **wrap_content** גובה האלמנט מינימלי, עוטף את תוכן האלמנט ותו לא.
match_parent גובה האלמנט מקסימלי, עד כמה שאלמנט האב מאפשר.
Height: **wrap_content** רוחב האלמנט מינימלי, עוטף את תוכן האלמנט ותו לא.
match_parent רוחב האלמנט מקסימלי, עד כמה שאלמנט האב מאפשר.
וכדומה...

קישור ג'אווה ו-XML

עם יצירת פרויקט האנדרואיד החדש, קבצי אקטיביטי (xml) ומיין (java) הדיפולטיביים כבר מכילים קישור זה לזה בביטוי- setContentView- נראה בסוגריים את שמו של קובץ ה- XML-
חשוב- במידה ויצרנו קובץ XML חדש חובה לזכור לכתוב אותו בביטוי זה, כדי לאפשר את הדיבור בין הג'אווה ל-XML.

יחס בין גבולות האלמנטים
Pedding- מרחק האלמנט מהגבולות של עצמו. ברירת מחדל פדינג 8
ניתן להגדיר כל גבול בנפרד- למעלה, למטה, שמאל, ימין.
Margin- מרחק גבולות האלמנט מאבא/ שכנים.
ניתן להגדיר כל גבול בנפרד- למעלה, למטה, שמאל, ימין.

מידות:

px- הרזולוציה קובעת את מספר הפיקסלים. **בחיים לא !!!!**

dp- מידה יחסית indepented pixel מחלק את הרוחב ל-160 px אינה מושלמת ביחס בגלל השארית. בא לידי ביטוי בסוג המכשיר. מידות אורך רוחב לאנדרואיד שהוא יחסי לרוחב המסך. לכן רלוונטי לאלמנטים. בכל זאת נשתמש בפרנטס ובשני.

sp- scael pixel ביחס לגודל הגופן ולא ביחס למסך! קיים הבדל בברירת המחדל של גודל גופן. נניח וברירת מחדל היא 8= בסלולרי הוא יהיה גדול יותר מאשר על הטבלט בברירת מחדל שלו כי הוא מותאם. כדי לתת פרופורציה לסוג המכשיר. רלוונטי לטקסט.

textSize- גודל הטקסט.
onClick- נוסיף בתוך הקלאס פונקציה שנקראת:
Public void funcOne(View v)- השימוש בה מחייב ייבוא של View v.
עכשיו נזהה text View v find by ID