

### Sql – lesson3

יש אפשרות למשוך את העמודה עפ"י השם שלה ולא עפ"י האידקס:

```
book = new Book (  
    c.getInt(c.getColumnIndex("_id")),  
    c.getString(c.getColumnIndex("name")),  
    c.getString(c.getColumnIndex("author")),  
);  
Return book;
```

### שלבי עבודה:

בכל כניסה ל onactivityResult יש הצגה של כל המערך. נגדיר בתוכו 3 דברים:

1. רשימה
  2. ליסט וויו
  3. אדפטר
- הצג רשימה.

עכשיו יש שינוי-

רשימה handler.getAllBook =

הפונקציה היא במקום רשימה ומחזירה כמערך.

-onCreate

1. רשימה handler.getAllNotes =

2. listView

3. להציג רשימה.

ארגז כלים:

addNote()

deleteNote()

updateNote()

getAllNotes()

getNote()

### sharedPreferences = שיתוף העדפות

לרוב אפליקציה מכילה מספר דפים, מה שמתבטא במספר ה-activity של האפליקציה, כל אחד מה-activity מתנהג כאפליקציה נפרדת (התנהגות חדשה שנועדה לחסכון בזיכרון האנדרואיד)

-Intent מקום שמור לאורך כל חי האפליקציה.

-DB תמיד נשאר, הוא כבד ומחזיק את כל הרשומות. מיקום: מחוץ לאפליקציה.

לפעמים יש ערכים שצריך לשימוש חד פעמי – למשל- שם משתמש וסיסמה, וזה לא מצריך טבלה.

כאן נשתמש בפונקציה זו: **SharedPreferences**

לקריאה נוספת <http://developer.android.com/reference/android/content/SharedPreferences.html>

זהו אובייקט שנועד לשמירת נתונים במבנה של key & value נעדיף מנגנון זה על פני DB כאשר

מדובר בנתון בודד לשימוש חד פעמי **לעומת** כמות נתונים מורכבת שיושבת בטבלה.

שומר נתונים פרימיטיביים בלבד: בוליאני, עשרוני, אינטגר, לונג ומחרוזת.

המידע מאוחסן בתיקיית ברירת מחדל או שהמתכנת יכול לציין שם תיקייה ספציפי שישמש לאחסון.

ה- sharedPreferences מנוהל בעזרתה של המתודה getSharedPreferences() השייכת לקלאס הקונטקסט.

### הדגמה: קביעת שם תיקייה ספציפי לאחסון הנתונים:

```
private static final String PREF_FILE_NAME="PrefFile";
SharedPreferences preferences = getSharedPreferences (PREF_FILE_NAME,MODE_PRIVATE);
במידה וקובץ ה-XML קיים – הוא נפתח. במידה והוא איננו קיים, הוא נוצר!
```

MODE\_PRIVATE – זוהי הדרך לתפעול ההעדפות, זוהי ברירת המחדל שהמשמעות שלה היא שהגישה ניתנת לאפליקציה קראה לה בלבד. (קיימות הרשאות נוספות אך מתברר שהן מהוות כפתח לפרצות אבטחה רבות ודרך ניתן להגיע לנתונים אישיים של המשתמש, הרשאות אלו לא מומלצות והן בדרך החוצה).

הדרך הנפוצה יותר היא להשתמש בקובץ הדיפולטיבי (ברירת המחדל) ולא לציין קובץ ספציפי

### הדגמה: שימוש בתיקייה דיפולטיבית לאחסון הנתונים:

```
SharedPreferences preferences = PreferenceManager.getDefaultSharedPreferences(this);
המשמעות של this – זהו ההקשר למציאת תיקיית ברירת המחדל עבורו.
המתודה תשתמש בשם ה- package כדי לקבוע את שם התיקייה ואת המיקום של תיקיית ההעדפות.
```

לרוב נשתמש בתיקיית ברירת המחדל, אלא במקרים שיש ריבוי משתמשים באפליקציה על מכשיר אחד וכל משתמש מנהל את ההעדפות שלו בעצמו – אז נשתמש באפשרות הראשונה כדי לשמור על הפרדה בין העדפות המשתמשים אחד מהשני.

### שליפת ההעדפות

לאחר הזנת הנתונים בפעם הראשונה, ניתן לשלוף אותם, תוך שימוש באחת המתודות הבאות:  
getInt(), getString(), getBoolean(), וכו' = תלוי בטיפוס הנתונים כמובן!  
דוגמאות:

```
String username = preferences.getString("userName","not-exist");
boolean opt = preferences.getBoolean("key", true );
```

### הסבר:

נשלוף את ההעדפה (=נתון, ערך) לתוך משתנה מאותו טיפוס. בסוגריים נרשום את שם המפתח הרצוי = המשתנה ונצפה שישלוף את ערך המשתנה. החלק השני של אותם סוגריים יהיה ערך ברירת מחדל שקובע המתכנת, שיוצג במידה ולא קיים משתנה זה.

### עריכת ההעדפות - השלבים:

1. נזמן את המתודה edit() כדי לקבל את עורך "שיתוף ההעדפות": SaredPreferences.Editor
  2. הוספה או עדכון של ערך ע"י שימוש באחת המתודות: putString(), putBoolean()
  3. שמירת הערך החדש באמצעות השיטה – commit()
- הדגמה: (כל שלב בשורה נפרדת)

```
SaredPreferences.Editor editor = settings.edit();
editor.putString("userName","not-exist");
editor.commit();
```

השימוש בפונקציה זו יוצר תיקייה בארגז החול ארגז החול אינו חלק מה- APK ויש לו את ההרשאות שלו.

## Styles

ה- style הוא אוסף של תכונות המסך (view) שניתן לתת לאוסף התכונות שם ולהשתמש בו כסט במגוון מסכים (views) הסטייל יכול לפרט מאפיינים כמו: גובה פדינג, צבע הכתב, גודל הפונט, צבע רקע ועוד הרבה יותר. הסטייל נמצא בקובץ XML נפרד מה-XML שמפרט את ה-layout. על מנת ליצור סט של סטייל – יש לשמור קובץ XML ב- <res <---values ישירות בפרויקט.

res --- values --- styles.xml

כל תכונות העיצוב נרשמות בקובץ זה.  
ניתן שם לסטייל –

ובתוכו תגית אייטם שם – ובין זה לתגית סגירה נגדיר ערך.  
בדרך זו ניתן לתת סט של הגדרות - מקביל ל- css3 inline

כאן ניתנה גם הגדרה שנקראת "ירושה" – אומר שאם יש סטייל שניתן לו שם כלשהוא, יש אפשרות לפתוח עוד סטייל שהשם שלו יהיה כלשהוא ונוסיף לו בתגית:

parent = "firstStyleName"

הוא מקבל את כל הגדרות העיצוב שנקבעו בסטייל הראשון וכאן מאפשר להוסיף עליהם הגדרות עיצוביות נוספות.  
הפריט יקבל את כל ההדרת הקודמות פלוס ההגדרות החדשות.

אפשר ללכת למניפסט, לאתר שם את האקטיביטי השני שלי – ושם בתגית של סקונד להגדיר-  
Android:theme="@style/newStyle"  
זה אמר אפשר לשם ברמת האפליקציה, ברמת העמד וכו

עד לפני שנה לא היה סטייל בכלל באקליפס בגלל החסכון בסוללה  
היום כבר יש סטיילים.

נוספו סטיילים מ-API ל-API כשמסתכלים קדימה.  
אך במבט אחורה זה לא יתפוס אם המינימום גרסא לא תופסת. וצריך לטפל בפתיחת האפליקציה בגרסאות ישנות.  
הטיפול בזה- חלוקה לתיקיות לפי גרסאות:  
נמצא תיקיות values ושם כל מיני תיקיות.  
Values-v14 וכדומה.

יש גרסאות API מ-8 ועד 19 (kitkat=) מגרסאות 14 ועד 19 נוספו סטיילים חדשים ובעצם לכל קבוצת גרסאות.  
מה נעשה אם משהו פותח בגרסה מאוד ישנה שלא תומכת בעיצוב.  
אנדרואיד מטפל בזה:

values- גרסאות 8 עד 10

values-v11- גרסאות 11 עד 13

values-v14- גרסאות 14 עד 19

המנגנון מספק את הסטייל המיטבי לכל גרסא.

**יש שאלה עליו בבחינה!**

values-sw600dp – טבלטים 7 אינץ'

ועוד אחת עם 720- טלטים 10.1 אינץ'

אפשר להשתמש במילה שמורה large ואז הוא פונה לכל הטבלטים.

LandPortExamble

(Ctrl f12 משנה את המצג)

res --- layout-land---

כשנכנסים לליאאוט און קרייט רץ ודבר ראשון הוא בודק אוריינטציה ואז מחפש תיקייה מתאימה –  
landאו port הוא נכנס לתוכו וחפש XML מתאים.  
אפשר להוסיף את תיקייה-  
dp600Sw ונשים בפנים XML ואם אין הוא ילך לליאאוט.  
כלומר לליאאוט הא ברירת מחדל- האנגרואיד מאתר סוג מכשיר ומצג מסך ולפי זה הוא הולך  
לתיקייה מתאימה שמכילה את העיצוב.  
אם אין הוא ילך לליאאוט שהא הברירת מחדל.

מתי זה עוזר כל הבלאגן הזה??  
יש נוסחא למספר dp וכך מתאפר לפתח תיקייה מותאמת למכשיר  
יש large שמתאים להכול.

landscape = קובע מצג קבוע: אורך או רוחב.  
דבר ראשון- נבדק גודל מסך  
דבר שני- אוריינטציה  
דבר שלישי- דיפו. ליאוט רגיל.  
שפה- ברירת מחדל הלוקל של המכשיר. אם לא הוא מחפש תיקיית וליוס עם הבחירה.