

Wygląda prosto? Poznaj i zakochaj się w świetnej alternatywie przechowywania informacji pomiędzy uruchomieniami aplikacji.

Aby w przechwycić, pobrać lub zapisać dane w trakcie wykonywania się aplikacji, ale nie tylko pomiędzy aktywnościami, ale stale przy każdym kolejnym uruchomieniu się aplikacji, najczęstszymi rozwiązaniami są SharedPreferences lub SQLite , jednak gdy tylko zaczniemy czytać w jaki sposób je wykorzystać lub jeżeli już dobrze je znamy , nie są to szczególnie dla początkujących najkrótsze prz najprostsze rozwiązania . ( tu rozwijacz )

A co jeśli można byłoby tak w trzech linijkach kodu - STWORZYĆ, ZAPISAĆ I ODCZYTAĆ dane? Brzmi pięknie , nieprawdaż ? To właśnie umożliwia nam klasa TinyDB.

Największa zaletą jej prostoty, a zatem rozwiązaniem na problem który ja szukałam jest szybki i banalny zapis danych takich jak obiektów, tablic obiektów. Podczas gdy przy używaniu SharedPreferences musimy iterować po tablicy, parsować obiekty na Jsony ( tak naprawdę to właśnie za nas robi ta napisana klasa) . ( tu konic rozwijacza)

A więc :

**TinyDB tinydb = new TinyDB(this); // tworzymy baze**

**tinydb.putInt("wartosc\_klucza",6); //zapisujemy w bazie wartosc 6 pod nazwa wartosc\_klucza**

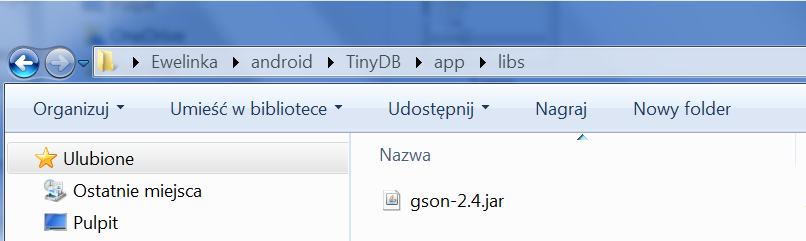
**int wartosc = tinydb.getInt("wartosc\_klucza"); //pobieramy za pomoca nazwy klucza zapisana wartosc**

**Inne możliwe typy do zapisu ( pobieranie analogicznie jak powyżej) :**

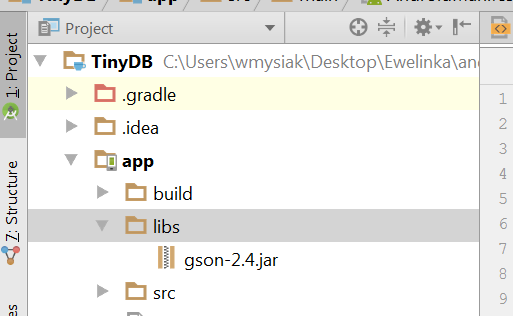
tinydb.putFloat("jakasNazwaKlucza", 6.6f);  
tinydb.putLong("x", 66666L);  
tinydb.putString("y", "Tak naprawdę to koty zawsze mają Ale ");  
tinydb.putBoolean("z", true);   
tinydb.putList("key", jakaśTablica);  
tinydb.putImagePNG("zamek", "zamek.png", lunchBitmap);  
tinydb.putObject("zamek\_blyskawiczny",jakisObiekt);  
tinydb.putListObject("a",jakasListaObiektow);

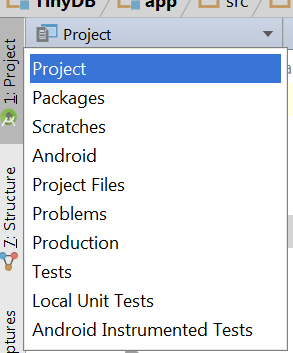
3 kroki które pozwolą Ci używać tej klasy :

1. Stwórz klase TinyDB i skopiuj zawartość : <https://github.com/kcochibili/TinyDB--Android-Shared-Preferences-Turbo/blob/master/TinyDB.java> odkomentuj w niej funkcje takie jak putObject.
2. <http://repo1.maven.org/maven2/com/google/code/gson/gson/2.4/> sciągnij plik json-2.4.jar
3. Pobrany plik wklejamy do folderu libs.



3.Otwieramy drzewo naszego projektu w AndroidStudio klikamy prawym przyciskiem na gson-2.4.jar a następnie klikamy opcje Add Lib.



Musimy pamiętać ze musimy mieć wybraną opcje drzewa wszystkich plików w projekcie, gdyż inaczej nie znajdziemy w nim folderu libs.