

Exp NO.1 : Introduction to Android Programming

* اريد أنواع ال OS يلي بنطورها هيا:

1) iOS (Apple)

2) Android (Google)

→ الـ Open Source

→ الـ Java لغة البرمجة

* لست بتقدر development الـ Android

هناك عتبات برمجية ios لازم اشتغل في بيئته الـ mac فقط

الـ ios اشتغل في بيئته الـ mac

الـ ios اشتغل في بيئته الـ mac

الـ ios اشتغل في بيئته الـ mac

* الـ ios اشتغل في بيئته الـ mac

* الـ ios اشتغل في بيئته الـ mac

Encapsulation ③

Abstract ②

Inheritance ①

polymorphism ④

Inheritance ①: الـ ios اشتغل في بيئته الـ mac

الـ ios اشتغل في بيئته الـ mac

الـ ios اشتغل في بيئته الـ mac

الـ ios اشتغل في بيئته الـ mac

B extends A

Abstract ← Interface ← Abstraction ②
method
Interface Type

Override ← Polymorphism ③
method
parent
implementation
A B
A B

Encapsulation ④
private
public
getter
setter

** Activity Classes

interface
Data
Data

** Services

Activity
Background
Application
services
Activity

** Broadcast Receiver

هو يشغل زِي interrupt وبنه بتتقد اجرت بعين بربوط
يكود وبتستقبل Analsment الو من اسيا و تا بنه كفا ينسد
هو الكود ، زِي فلكا تا يذبح الفيت مع الفيت بيطل في connection
على ال Facebook ، اى موقع توصل تا في

* Content Providers

هو د اوف كيز بي اسكرم ال Component بيت ال method تا عفا
ال Application تا بنه ، مثلاً بي اخل ال email مستخدم
method اى عن طريق ال Content providers

** Android Manifest File

هو ال الفاي عبارة عن (xml) قابل ، اجنوب بك جميع ، كنها زهد
وال Activity و ال (3) هوراك ال Application بيتي و كل سني
هو موجود به (ما صنفه كل ، محتويات ال البرنامج ال Application)

** R. Java Class

عبارة عن static كلاس ، بيت بعن المعلومات عن ال id اى
موجودة لاني Button ، اى text اى ايني وبتسوف بعن
لاني ال Component ، موجودة كذا

Exp NO2 : Android layouts.

* هذه التجربة توضح كيف نعمل أنواع ال layouts في ال Application
وكيف نعمل ال layouts ونعمل ال linear layout
سواء ال vertical أو ال Horizontal ، وكيفية ال Constrain layout
أو ال relative نفس الطريقة ، وكيفية ترتيب ال elements
وكيفية نقل ال Activity جديدة عن ال main Activity ونقل ال

* ال Android في عنا نوعين من ال views :-

① Widgets → ال user يتفاعل مع ال
شكل مباشر ، وما يلاحظ فيه ال Text Field ، Buttons
table ... etc ، ال Review ال extends ال view class

② Layouts → ال UI element على الواجهة يعني
ال UI element : كل نوع معين ، وطار للوضع الـ حسب نوع
ال UI element ، يعني سواء كان ال (Button ، Text ... etc)
وفيها عدة أنواع لنقل ال extends ال view group class ،
ال view group class ال extends ال view class

* أنواع ال layouts :-

① Linear Layout → A- Vertical

B- Horizontal

② Relative Layout → ال relative ال extends ال view group class
أو ال extends ال view class ، ال child ال extends ال view class

③ Web view

* في عري (تخطيط) بال layout - height

① match parent → هذا يعني اننا نساوي الارتفاع

② wrap parent → هذا يعني اننا نساوي العرض

** لو ضلنا عري في element او جوار Horizontal layout عري

او طولهم نفس الحجم . اما بعرض او كبر او اقل

* match parent بعرض او كبر او اقل

او طولهم نفس الحجم . اما بعرض او كبر او اقل

** او Spinner ← تقريباً مثل ال Combo Box

** بعرض او كبر او اقل layout جوار layout بعرض او كبر او اقل

① Wide → هذا يعني اننا نساوي العرض

② Deep → هذا يعني اننا نساوي الارتفاع

← هذا يعني اننا نساوي العرض او الارتفاع

loading في ال view هذا يعني اننا نساوي العرض

one layout بعرض او كبر او اقل

او طولهم نفس الحجم . اما بعرض او كبر او اقل

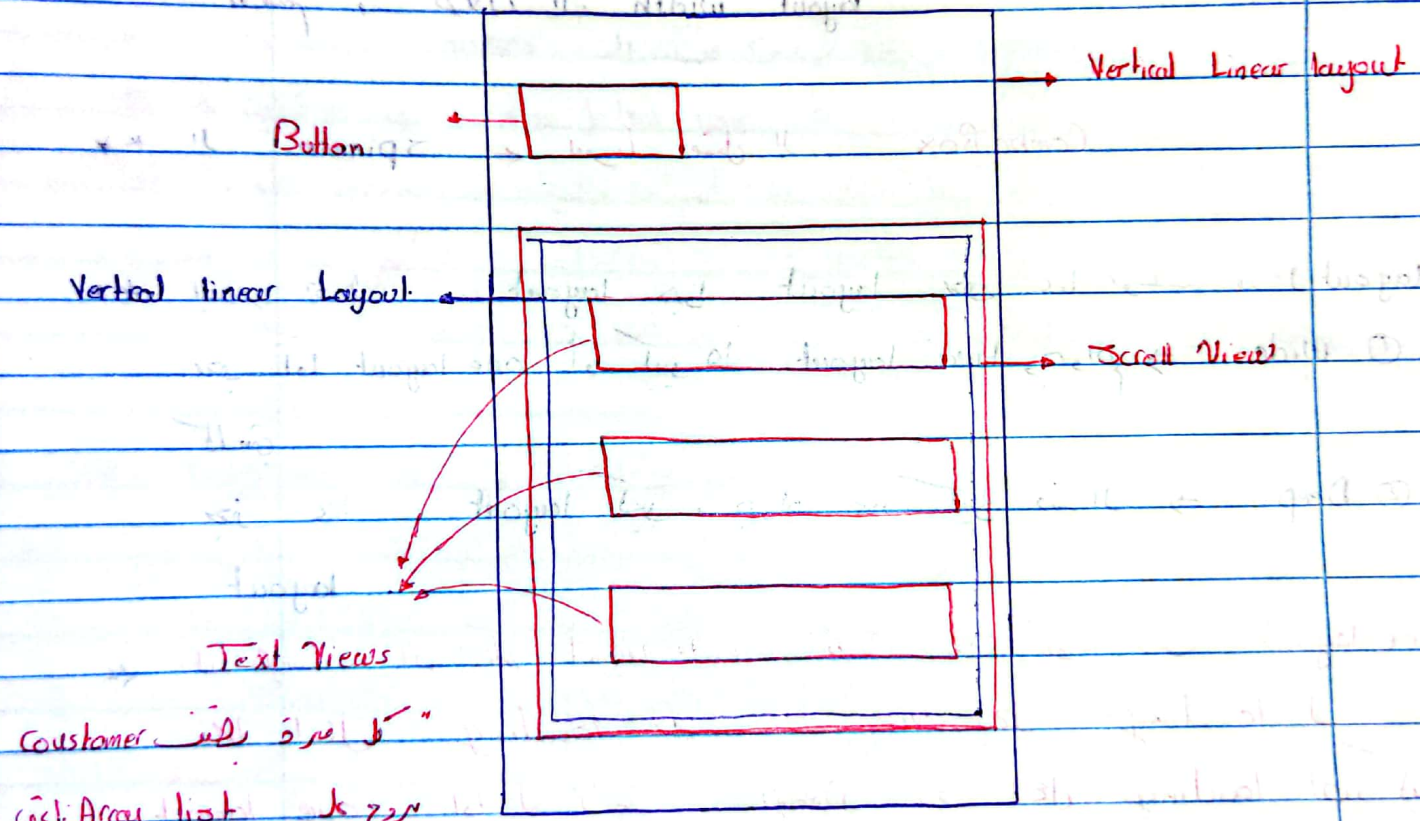
* هذا يعني اننا نساوي العرض او الارتفاع

← هذا يعني اننا نساوي العرض او الارتفاع

XML

* كَيْفَ يَرَى اَصْحَابُ اَلْمَرَّةِ اَلْمَرَّةَ اَلْمَرَّةَ بِرَجُلٍ
 اَنْتَظِمُ؟ كَمَا اَلْعَلِيَّةُ اَنْتَ اَلْمَرَّةَ اَلْمَرَّةَ
 مِنْ خَلْفِ اَلْمَرَّةِ اَلْمَرَّةِ اَلْمَرَّةِ اَلْمَرَّةِ
 اَلْمَرَّةِ اَلْمَرَّةِ اَلْمَرَّةِ اَلْمَرَّةِ اَلْمَرَّةِ
 اَلْمَرَّةِ اَلْمَرَّةِ اَلْمَرَّةِ اَلْمَرَّةِ اَلْمَرَّةِ

* Description :-



" كَمَا اَلْمَرَّةَ اَلْمَرَّةِ اَلْمَرَّةِ
 اَلْمَرَّةِ اَلْمَرَّةِ اَلْمَرَّةِ اَلْمَرَّةِ
 اَلْمَرَّةِ اَلْمَرَّةِ اَلْمَرَّةِ اَلْمَرَّةِ

كَمَا اَلْمَرَّةَ اَلْمَرَّةِ اَلْمَرَّةِ interface كَمَا اَلْمَرَّةِ

" Dynamic " كَمَا اَلْمَرَّةَ اَلْمَرَّةِ Java Code كَمَا اَلْمَرَّةِ

* اَلْمَرَّةِ اَلْمَرَّةِ اَلْمَرَّةِ اَلْمَرَّةِ a XML كَمَا اَلْمَرَّةِ اَلْمَرَّةِ اَلْمَرَّةِ a Static كَمَا اَلْمَرَّةِ اَلْمَرَّةِ اَلْمَرَّةِ اَلْمَرَّةِ

(XML) كَمَا اَلْمَرَّةِ اَلْمَرَّةِ

* کتب اعلیٰ Activity جیسے ہے
 ← وہی اعلیٰ Model class اسے Customer، Customer، جو ال Customer
 نظر مجموعہ Info، Id، name، etc
 حوالہ دینے کے لیے ArrayList
 عنوان کے ساتھ ساتھ Customer کے لیے اس کے لیے
 ال (list)

← وہی اعلیٰ new Activity جیسی ہے
 ① ال سب سے آسان طریقہ (Simple way) ہے، عنوان کے لیے ای
 Activity، ال Activity کے لیے Java Code، XML ان کے لیے
 Manifest

② Standard ← ہرگز File، new ← Java Class
 Class name کے ساتھ ساتھ Capital، ال کے ساتھ ساتھ ال Class
 extends of AppCompatActivity، ال Override، OnCreate
 کے ساتھ ساتھ ال Manifest File کے ساتھ ساتھ ال کے ساتھ ساتھ ال

```
<activity android:name=".SecondActivity">
```

* ال کے ساتھ ساتھ ال Second کے ساتھ ساتھ ال (xml)

ہرگز کے ساتھ ساتھ ال کے ساتھ ساتھ ال Second Activity کے ساتھ ساتھ ال
 super, onCreate, ...

```
setContentView(R.layout.activity_second);
```

* ال کے ساتھ ساتھ ال ال کے ساتھ ساتھ ال Right Click کے ساتھ ساتھ ال ال کے ساتھ ساتھ ال main
 ← new ← Activity ← Empty Activity، ال کے ساتھ ساتھ ال ال کے ساتھ ساتھ ال ال کے ساتھ ساتھ ال
 ال XML Code

* ايسٲ ميٲى **Constrain layout** ؟
هو ايسٲ ميٲى ال **Relativ** لس ال **Constrain** اٲ ءبر ، لس
اٲط نىو **Button** نىو ، لس لس سبب علٲو error فى
المامو ، لانا ما نعرف مٲ رءىا عن لسٲى لانا ما اٲرء ال اٲار
لسٲى سو ءءو و لس ما بوى .

ءرف سو ءءو سو ءءو ال (xml) كود ، و لسٲى ، لسٲى ا ٲٲ
Text view ، اٲكم سو ءءو لسٲى اٲرءىا

* لسٲى انءزل مٲى **Activity** اٲ **Activity** لسٲى بسءى بسءى ؟

* Type of intents

① Explicit intents → احنا بنوفر ال intent ال Application بي بي

أقولها بال name component

② Implicit intents → احنا ما بنسبى ال component اسم بالزبط

بنطرحها Action معينة، فالتطبيق بي اطلع اني يعرف ال Action معينة

نفسه اني بقولها ال Action، فالتطبيق بي اطلع ال Application بي اطلع ال Action

← ال Action بي اطلع ال Application بي اطلع ال Action

ال Action بي اطلع ال Application بي اطلع ال Action

ال Action بي اطلع ال Application بي اطلع ال Action

ال Action بي اطلع ال Application بي اطلع ال Action

* User Notification

→ Two Type: ① Toast Message : عبارة عن رسالة تظهر على الشاشة لفترة قصيرة ثم تختفي

ال رسالة تظهر على الشاشة لفترة قصيرة ثم تختفي

ال رسالة تظهر على الشاشة لفترة قصيرة ثم تختفي

② Notification Area (status Bar Notification) :

هنا بيطلع على التلوين من فوق يعني لو تيركت الشاشة بيطلع عليها

ال استعارات و بتقدر تطرحها remove (بيبي بي اطلع ال User ال User)

استعارات ال Facebook , messenger ... etc

* Notification

① أدار طريقة لازم اطلع اني اسمو channel ، بيطلعها من خلال

object اسمو Notification channel ، و بتطرحها مختلف ال ال ال

① Id and name channel

② Importance → channel ال ال ال

7. importance ال **importance** ال **Android setup** ال **User** ال **نيل** ال
نيل ، **remove** ال **نيل** ال **نيل** ال **نيل** ال

2. ال **notification** ال **channel** ال **create** ال **نيل** ال **نيل** ال
ال **Notification Compat** ال **نيل** ال **نيل** ال **نيل** ال
ال **Builder** ال **نيل** ال **Object** ال **نيل** ال **نيل** ال
Object ال **نيل** ال **set** ال **نيل** ال **نيل** ال **Builder** ال
ال **NotificationCompat** ال **نيل** ال **نيل** ال **نيل** ال
ال **build** ال **نيل** ال **نيل** ال **نيل** ال **نيل** ال
ال **notification** ال **نيل** ال **نيل** ال **نيل** ال
ال **notification** ال **نيل** ال **نيل** ال **نيل** ال

ال **Intent** ال **نيل** ال **نيل** ال **نيل** ال **نيل** ال
ال **Pending Intent** ال **نيل** ال **نيل** ال **نيل** ال **نيل** ال
ال **Activity** ال **نيل** ال **نيل** ال **نيل** ال **نيل** ال
ال **notification** ال **نيل** ال **نيل** ال **نيل** ال **نيل** ال
ال **notification Action** ال **نيل** ال

Experiment NO.4 : SQLite Database

Main Activity کا رخ Exp2 کے تجربے کے ساتھ کیا گیا۔

Add Customer Act.

Main Activity کا رخ Add Customer Act. کے ساتھ کیا گیا۔

Customer list کے تجربے کے ساتھ کیا گیا۔

Customer class

Customer list کے تجربے کے ساتھ کیا گیا۔

locally SQLite File Database کے تجربے کے ساتھ کیا گیا۔

device کے تجربے کے ساتھ کیا گیا۔

SQLite Database اور Array list کے تجربے کے ساتھ کیا گیا۔

Application کے تجربے کے ساتھ کیا گیا۔

SQLiteOpenHelper class کے تجربے کے ساتھ کیا گیا۔

Android کے تجربے کے ساتھ کیا گیا۔

کتاب کے تجربے کے ساتھ کیا گیا۔

SQLiteOpenHelper class کے تجربے کے ساتھ کیا گیا۔

اس کے تجربے کے ساتھ کیا گیا۔

SQLiteOpenHelper class کے تجربے کے ساتھ کیا گیا۔

getWritableDatabase() کے تجربے کے ساتھ کیا گیا۔

getReadableDatabase()

نہیں تاکہ میٹریڈ اس insert() ، اڈا اس کے لئے تمام بچل
insert methods اس Database ، اڈا اس کے لئے برعکس (-1) ،

اڈا اس کے لئے اڈا اس کے لئے اڈا اس کے لئے اڈا اس کے لئے اڈا اس کے لئے
methods اس کے لئے اڈا اس کے لئے اڈا اس کے لئے اڈا اس کے لئے اڈا اس کے لئے
اس کے لئے اس کے لئے اس کے لئے اس کے لئے اس کے لئے اس کے لئے اس کے لئے اس کے لئے

اس کے لئے اس کے لئے اس کے لئے اس کے لئے اس کے لئے اس کے لئے اس کے لئے اس کے لئے
Attributes اس کے لئے اس کے لئے اس کے لئے اس کے لئے اس کے لئے اس کے لئے اس کے لئے اس کے لئے
Database اس کے لئے اس کے لئے اس کے لئے اس کے لئے اس کے لئے اس کے لئے اس کے لئے اس کے لئے

اس کے لئے اس کے لئے اس کے لئے اس کے لئے اس کے لئے اس کے لئے اس کے لئے اس کے لئے
onResume() اس کے لئے اس کے لئے اس کے لئے اس کے لئے اس کے لئے اس کے لئے اس کے لئے اس کے لئے
onCreate() اس کے لئے اس کے لئے اس کے لئے اس کے لئے اس کے لئے اس کے لئے اس کے لئے اس کے لئے

Procedure اس کے لئے اس کے لئے اس کے لئے اس کے لئے اس کے لئے اس کے لئے اس کے لئے اس کے لئے

اس کے لئے اس کے لئے اس کے لئے اس کے لئے اس کے لئے اس کے لئے اس کے لئے اس کے لئے
class اس کے لئے اس کے لئے اس کے لئے اس کے لئے اس کے لئے اس کے لئے اس کے لئے اس کے لئے
extends اس کے لئے اس کے لئے اس کے لئے اس کے لئے اس کے لئے اس کے لئے اس کے لئے اس کے لئے
Override اس کے لئے اس کے لئے اس کے لئے اس کے لئے اس کے لئے اس کے لئے اس کے لئے اس کے لئے
Constructor اس کے لئے اس کے لئے اس کے لئے اس کے لئے اس کے لئے اس کے لئے اس کے لئے اس کے لئے
Object اس کے لئے اس کے لئے اس کے لئے اس کے لئے اس کے لئے اس کے لئے اس کے لئے اس کے لئے

اس کے لئے اس کے لئے اس کے لئے اس کے لئے اس کے لئے اس کے لئے اس کے لئے اس کے لئے

اس کے لئے اس کے لئے اس کے لئے اس کے لئے اس کے لئے اس کے لئے اس کے لئے اس کے لئے
onCreate اس کے لئے اس کے لئے اس کے لئے اس کے لئے اس کے لئے اس کے لئے اس کے لئے اس کے لئے
database اس کے لئے اس کے لئے اس کے لئے اس کے لئے اس کے لئے اس کے لئے اس کے لئے اس کے لئے

اس کے لئے اس کے لئے اس کے لئے اس کے لئے اس کے لئے اس کے لئے اس کے لئے اس کے لئے
UpCreate اس کے لئے اس کے لئے اس کے لئے اس کے لئے اس کے لئے اس کے لئے اس کے لئے اس کے لئے
Feature version , current version اس کے لئے اس کے لئے اس کے لئے اس کے لئے اس کے لئے اس کے لئے اس کے لئے اس کے لئے

DatabaseHelper class mein insertCustomer() method ko likhna hai jo database mein insert karne ke liye use ki jayegi. DatabaseHelper class mein insertCustomer() method ko likhna hai jo database mein insert karne ke liye use ki jayegi.

getAllCustomer() method ko likhna hai jo database se sabhi customer ko fetch karne ke liye use ki jayegi. Cursor ko use karke database se data fetch karne ke liye use ki jayegi.

ArrayList ko use karke database se data ko store karne ke liye use ki jayegi. ArrayList ko use karke database se data ko store karne ke liye use ki jayegi.

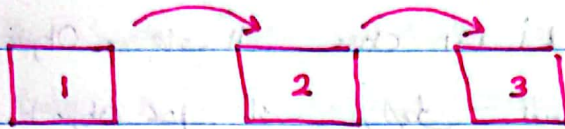
create() method ko likhna hai jo database mein create karne ke liye use ki jayegi. DatabaseHelper class mein create() method ko likhna hai jo database mein create karne ke liye use ki jayegi.

Override OnResume() method ko likhna hai jo database se data ko fetch karne ke liye use ki jayegi. OnResume() method ko use karke database se data ko fetch karne ke liye use ki jayegi.

delete() method ko likhna hai jo database se delete karne ke liye use ki jayegi. DatabaseHelper class mein delete() method ko likhna hai jo database se delete karne ke liye use ki jayegi.

* Experiment NO. 5 Animation and Tween Animation in Android

① Frame Animation: يكون على هيئة صور كل صورة بناخذ time معنا
وحدها يظهر أو تختفي بين كل صورة وصورة وهكذا
"عند كلف أنتقل الفيديو كالتالي"



* الفرق بين الصور المتتالية بتغيير الأداة في صورة معينة للصور

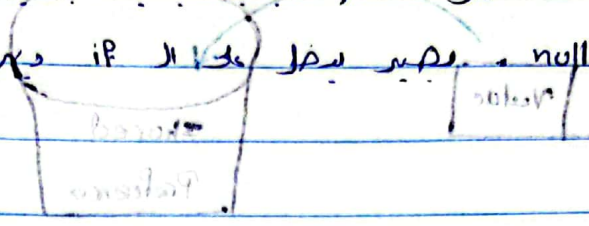
② Tween Animation: يكون على هيئة صورة واحدة بحركتها بال
Direction إلى أي جهة بالها تظهر الصورة وهكذا
① Feel in: تظهر الصورة
② Feel out: تختفي الصورة

* In Frame Animation we use AnimationDrawable
وهو المسؤول عن كل شيء في Animation

* In Tween Animation we use Animation in XML File
that performs (أي يكون على هيئة XML):
① Rotating ② Pading ③ moving ④ stretching

** Animation Class: Animation details في كل جزء Animation
في Animation Patch Animation في كل جزء Animation

هذا يعني أننا انما نستخدم الـ `getInstance()` الـ `Object` لو هو `null` نرفع
 نستخدمه اذا لزمنا نقل واحد من غيره بحيث نرجعها واحد الجيد ، بقاى طرق
 نستخدمه `getInstance()` . `getInstance()` الـ `IP` في الـ `getInstance()` نستخدمه ، هكذا



* * * الـ `shared preferences` ← عبارة عن `File` مخبأ ، هذا الـ `File` يكون
 مخبأ في ذاكرة الهاتف ، نستخدمه `save()` ، `load()` ، `edit()` ، `clear()` ، `removeAll()` ، `removeEntry()`

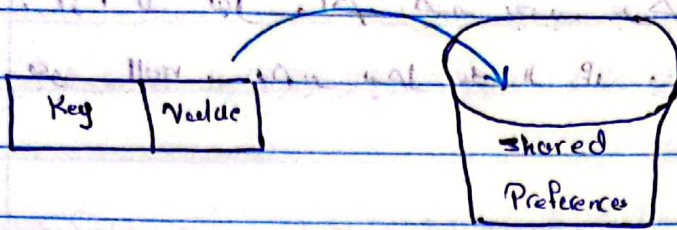
← الـ `Format` في الـ `File` في الـ `save()` : `XML` ، `JSON` ، `device`

← الـ `shared preferences` يقبل `primitive type` ، `String` ، `Boolean` ، `float` ، `long` ، `int` ، `String` .

* `method` في الـ `shared preferences class` :
 ① الـ `getPreferences()` ← يعطينا الـ `mode` الـ `private` ، `Axis` ، `Axis`
 ② الـ `getSharedPreferences()` : `mode private` ، `Axis`
 ③ الـ `getSharedPreferences()` : `mode world readable` ، `Axis`
 ④ الـ `getSharedPreferences()` : `mode world writable` ، `Axis`

Only One preference File في الـ `shared preferences` ، `mode`

⑤ الـ `getSharedPreferences()` : `mode` ، `Axis`



← Data Shared Key Value في Key Value جدول
 في Save في xml file او file انشور Shared Pref.

* Editor ← Editor هو الذي يكتب في ال Shared Preferences
 editor هو الذي يكتب في ال Shared Preferences
 editor هو الذي يكتب في ال Shared Preferences
 editor هو الذي يكتب في ال Shared Preferences
 editor هو الذي يكتب في ال Shared Preferences

* Commit في ال Shared Preference
 Commit في ال Shared Preference
 Commit في ال Shared Preference
 Commit في ال Shared Preference
 Commit في ال Shared Preference

* Database ← Key Value unique
 Database ← Key Value unique
 Database ← Key Value unique
 Database ← Key Value unique
 Database ← Key Value unique