

# DEPENDENCY INJECTION İLE LOCAL DATABASE

Yakup Okumuş

yakup2525



yakup\_okumus@outlook.com



yakupokumus.com



## 01 DEPENDENCY INJECTION (GET\_IT)

- Kullanım Kolaylığı
- Singleton Desteği
- Lazy Initialization
- Düzenli ve Test Edilebili Kod

## LOCAL STORAGE (SHARED PREFERENCES)

- ### 02
- Küçük Veriler
  - Basit Kullanım
  - Key-Value
  - Düşük Veri Boyutu

## 03 LOCAL STORAGE (HİVE)

- Hızlı ve Hafif
- Özelleştirilebilir Model Sınıfları
- NoSQL Veri Tabanı
- Güvenli



# SHARED PREFERENCES NEDİR

FLUTTER'DA KULLANILAN SHARED PREFERENCES, BASİT VERİLERİ SAKLAMAMIZI SAĞLAYAN KÜÇÜK BİR SAKLAMA MEKANİZMASIDIR, BU MEKANİZMA, KÜÇÜK MİKTARLARDA VERİYİ ANAHTAR-DEĞER ÇİFTLERİ (KEY-VALUE PAIRS) ŞEKLİNDE DEPOLAMAK İÇİN KULLANILIR. ÖZELLİKLE UYGULAMA AYARLARI, OTURUM BİLGİLERİ, TERCİHLER VEYA KÜÇÜK BOYUTLU VERİLER GİBİ BASİT VERİLERİ YEREL OLARAK SAKLAMAK İÇİN UYGUNDUR.CAL STORAGE) İÇİN KULLANILIR.

# HİVE NEDİR

FLUTTER'DA KULLANILAN HIVE, AÇIK KAYNAK BİR NOSQL TABANLI VERİTABANI PAKETİDİR. HIVE, HIZLI, HAFİF VE YEREL SAKLAMA (LOCAL STORAGE) İÇİN KULLANILIR.

# GET\_it NEDİR

GETIT, FLUTTER VE DART GIBİ DILLERDE SERVİSLERİN, BAĞIMLILIKLARIN VE NESNELERİN YÖNETİMİNİ KOLAYLAŞTIRMAK İÇİN KULLANILAN BİR BAĞIMLILIK ENJEKSİYONU (DEPENDENCY INJECTION) KÜTÜPHANESİDİR. GETIT, BAĞIMLILIKLARI ÇÖZMEK VE YÖNETMEK İÇİN BİR SERVİS KONFIGÜRASYONUNU KOLAYCA TANIMLAMANIZA OLANAK TANIR VE BU SERVİSLERİ UYGULAMA BOYUNCA KULLANABİLİRSİNİZ.

# GEREKLİ PAKETLER

pub.dev

- hive
- hive\_flutter
- hive\_generator
- build\_runner
- shared\_preferences
- get\_it



# YAPILMASI GEREKEN ÖNEMLİ ADIMLAR



## SHARED PREF MANAGER

Shared preferences ile yapacağımız işlemler için interface ve manager sınıf oluşturalım



## HIVE MANAGER

Hive ile yapacağımız işlemler için interface ve manager sınıf oluşturalım



## İNJECT

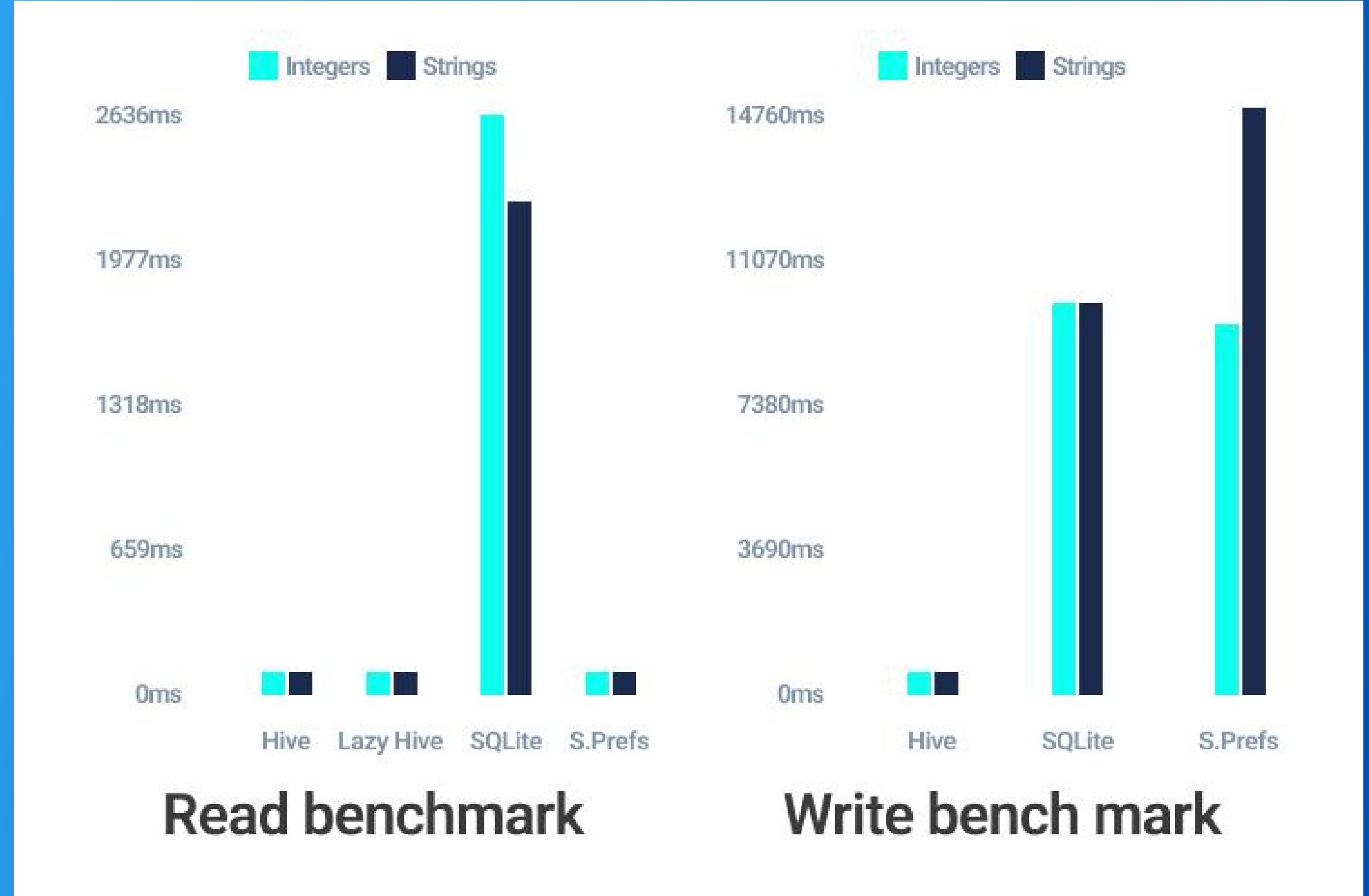
get\_it ile singleton oluşturup her yerde kullanalım

# BENCH MARK

Hive ve Shared Preferences hızlarını karşılaştıralım

## İSTATİSTİKLER

Okuma performansı söz konusu olduğunda SharedPreferences, Hive ile aynı seviyededir. SQLite çok daha kötü bir performans ile karşımıza çıkar. Hive, yazma veya silme konusunda SQLite ve SharedPreferences'tan büyük ölçüde daha iyi performans gösterir.





# TEMEL FOKSIYONLAR

## HİVE



- Hive.initFlutter()
- Hive.openBox(boxName)
- Hive.boxExists(boxName)
- Hive.deleteBoxFromDisk(boxName)
- \_box.add(value)
- \_box.put(key, value)
- \_box.get(key)
- \_box.delete(key)
- \_box.clear()
- \_box.close()

# TEMEL FOKSIYONLAR

## SHARED PREF



- setInt(key)
- setString(key)
- setBool(key)
- setDouble(key)
- getInt(key)
- getString(key)
- getBool(key)
- getDouble(key)
- remove(key)
- containsKey(key)
- clear()

# DEPENDENCY INJECTION(GET\_IT)

- initLocator()
- registerSingletonAsync

```
-getIt.registerSingletonAsync<IHiveManager>() async =>  
HiveManager().init());
```

```
getIt.registerSingletonAsync<ISharedPreferencesManager>  
(() async => SharedPreferencesManager().init());
```



# DIĞER DATABASE SEÇENEKLERİ



Isar Database



Object Box



SQLite



**realm**  
by MongoDB

Realm



# DINLEDİĞİNİZ İÇİN TEŞEKKÜRLER

---

yakupokumus.com