**5 approaches**

1. **Using the Shared Preferences**

**You can use the shared preferences if you only have a little amount of data to keep and don’t want to use the internal storage. Shared Preferences are used to store data in a key-value format, which means you’ll have one key and the associated data or value will be stored depending on that key. The data saved in the shared preferences will remain with the application until you delete it from your phone. All shared preferences will be deleted from the device if you uninstall the application.**

Pros: Easy to use and easy to understand. Very lightweight. Our app will store some user setting, so shard preferences is a good choice.

Cons: Can not do conditional query. Data type is restrained. It would be troublesome if our application were to expand in the future.

1. **Storage on the Inside**

**When you install an app on your phone, the Android operating system will give you some form of secret internal storage where the app can store its private data. No other application has access to this information. When you uninstall an application, all of the data associated with it is also removed.**

Pros: Easy to store some media data, such as our icon, the screen shot of the translated screen.

Cons: May cost a lot of memory space. So maybe we should clear our data at regular intervals.

1. **External Hard Drives**

**Most Android devices have relatively low internal storage. As a result, we keep our data on an external storage device. These storage units are accessible to everyone, which means they can be accessed by all of your device’s applications. You can also access the storage by connecting your mobile device to a computer. You must obtain the *READ EXTERNAL STORAGE* permission from the user in order to gain access to the external storage. As a result, any application with this permission has access to your app’s data.**

Pros: Can store a lot of media data.

Cons: I believe that very few of our target users use devices with SD cards today. Dealing with device compatibility issues can be complex.

1. **Using Android Database**

**Databases are collections of data that are organized and saved for future use. Using a Database Management System, you can store any type of data in your database. All you have to do is establish the database and use one query to perform all of the operations, such as insertion, deletion, and searching. The query will be passed to the database, which will return the desired output. In Android, an SQLite database is an example of a database.**

Pros: Easy to use and fully functional, but also has a good security.

Cons: More complicated than the first approach for our data. Our app can use database to store user information if we need. But shared preference is enough for now.

1. **Using Internet storage**

**Use network protocols to store data to the cloud**

Pros: No worried about the cost of memory and easy to handle data on service. Our app may use internet to store the VIP information.

Cons: There will be network latency issues and security issues. Our app will acquire a lot of permissions, so that using internet to transfer data may cause security concern.

Reference: [Storage System to Store Data in Android - GeeksforGeeks](https://www.geeksforgeeks.org/storage-system-to-store-data-in-android/)