

Week 9 note

- Android cung cấp các phương thức khác nhau để lưu trữ dữ liệu, tùy thuộc vào nhu cầu và mục đích của lập trình viên, ví dụ: lưu trữ các cài đặt, hay là ảnh, hoặc là dữ liệu cần sự bảo mật, ...
- Android cung cấp cơ chế để truy cập dữ liệu private : là ContentProvider, cho phép đọc/ghi dữ liệu trong một ứng dụng hoặc trong các thư mục mà được cho phép bởi Android

Các giải pháp lưu trữ dữ liệu

Shared preferences

- Lưu những dữ liệu dung lượng nhỏ, lâu dài như cài đặt hay trạng thái người dùng
- Tương tác với shared preferences như một dictionary: dữ liệu được định danh bằng các key kiểu string: data chứa các dữ liệu nguyên thủy như Boolean, int, float, long, ...
- Được lưu trong tệp XML trên thiết bị có đường dẫn :
/data/data/<app_package_name>/shared_prefs
- Bị xóa khi ứng dụng xóa
- Các cách lấy preferences:
 - + getSharedPreferences: cho phép truyền vào tên, sử dụng khi có nhiều tệp preferences xác định bởi tên
 - + getPreferences: sử dụng khi chỉ có 1 tệp preferences
- Các phương thức đọc tệp preferences:
 - contains(String key) - Kiểm tra xem preferences có chứa một preference cụ thể không
 - getAll() - Lấy toàn bộ values từ preferences
 - getBoolean(String key, boolean defValue) - Lấy một giá trị kiểu boolean từ preferences.
 - getFloat(String key, float defValue) - Lấy một giá trị kiểu float từ preferences.
 - getInt(String key, int defValue) - Lấy một giá trị kiểu int từ preferences.
 - getLong(String key, long defValue) - Lấy một giá trị kiểu long từ preferences.
 - getString(String key, String defValue) - Lấy một giá trị kiểu String từ preferences.
 - getStringSet(String key, Set<String> defValues) - Lấy một tập giá trị kiểu String từ preferences.

Android file system:

- Dùng để lưu các tệp tạm thời hoặc lưu trữ dữ liệu (CSV, JSON, ...)

Bộ nhớ trong

- Lưu các tệp tin vào trong bộ nhớ của thiết bị
- Các tệp tin này thường sẽ private, nghĩa là chỉ có ứng dụng của mình mới có thể truy cập chúng

- Sẽ bị xóa khi ứng dụng bị xóa
- Sử dụng `FileInputStream` để mở một tệp tin cho việc đọc
- Sử dụng `FileOutputStream` để mở một tệp tin cho việc ghi
 - + `MODE_PRIVATE`: chế độ mặc định, tạo các tệp tin private chỉ có ứng dụng bạn có thể truy cập. Sẽ xóa hết file
 - + `MODE_APPEND`: mở file sẵn có và tiếp tục ghi vào nó

Bộ nhớ ngoài

- Là các thiết bị lưu trữ có thể tháo bỏ (vd thẻ nhớ) hoặc các thiết bị lưu trữ cố định
 - Các tệp tin ở bộ nhớ ngoài world-readable: có thể sửa đổi bởi các yếu tố bên ngoài như máy tính đang kết nối với điện thoại hay các ứng dụng khác
 - Yêu cầu quyền truy cập hệ thống `READ_EXTERNAL_STORAGE` hoặc `WRITE_EXTERNAL_STORAGE`
 - Kiểm tra tính khả dụng của bộ nhớ ngoài sử dụng `getExternalStorageState`
 - Tệp tin công khai: luôn sẵn sàng được sử dụng bởi các ứng dụng khác
 - Sử dụng `getExternalStoragePublicDirectory(string type)` để nhận thư mục các tệp tin lưu trữ công khai
 - Tệp tin private: vẫn có thể được truy cập bởi chỉ riêng ứng dụng của bạn
- SQLite
- Lưu trữ các danh sách sản phẩm, thông tin người dùng hoặc lịch sử truy cập => những thứ gì có thể hệ thống hóa/cấu trúc được thành bảng các trường/cột
 - Android hỗ trợ sẵn SQLite
 - Lớp `Contract` chỉ định rõ khung của lược đồ dữ liệu theo cách có hệ thống và tài liệu hóa
 - + Tạo định nghĩa lớp có tính tổng hợp cho cả cơ sở dữ liệu
 - + Tạo inner class cho mỗi bảng liệt kê

Network

- Lưu các dữ liệu trên mạng/mây
- Tương tác sử dụng lớp `.net` của Java/Android