

CÀI ĐẶT ASYNCTASKLOADER

1. Tạo 01 lớp AsyncTaskLoader

- Tạo lớp **TapThoLoader** kế thừa từ lớp **AsyncTaskLoader** với kiểu tham số là **String**

```
public class TapThoLoader extends AsyncTaskLoader<String> {  
  
}
```

- Cài đặt phương thức khởi tạo (*ALT + ENTER: Create constructor matching supGọi*)

```
public TapThoLoader(@NonNull Context context) {  
    super(context);  
}
```

- Cài đặt phương thức **loadInBackground** (*ALT + ENTER: Implement methods*)

```
@Nullable  
@Override  
public String loadInBackground() {  
    return NetworkUtils.getJSONData("tap-tho", "GET");  
}
```

- Cài đặt phương thức **onStartLoading** (*Code > Override Methods...*)

```
@Override  
protected void onStartLoading() {  
    super.onStartLoading();  
    forceLoad();  
}
```

Gọi thực thi phương thức **forceLoad()** để bắt đầu thực thi phương thức **loadInBackground()**. *Loader sẽ không tải dữ liệu cho đến khi gọi phương thức **forceLoad()***

2. Cài đặt MainActivity

- Cài đặt kế thừa interface **LoaderManager.LoaderCallbacks** với kiểu tham số là **String**

```
public class MainActivity extends AppCompatActivity  
    implements LoaderManager.LoaderCallbacks<String> {  
  
}
```

- Cài đặt các phương thức từ interface **LoaderManager.LoaderCallbacks** (*ALT + ENTER: Implement methods*)

```

@NonNull
@Override
public Loader<String> onCreateLoader(int id, @Nullable Bundle args) {
    return new TapiThoLoader(this);
}

@Override
public void onLoadFinished(@NonNull Loader<String> loader, String data) {
    // Cài đặt tương tự như phương thức onPostExecute() trong AsyncTask
    // xử lý dữ liệu JSON (data)
}

@Override
public void onLoaderReset(@NonNull Loader<String> loader) {
}

```

Trong đó:

- **onCreateLoader():** được gọi khi khởi tạo loader.
 - **onLoadFinished():** được gọi khi task của loader đã hoàn thành. Viết code để cập nhật UI với kết quả trả về.
 - **onLoaderReset():** xóa toàn bộ tài nguyên còn lại.
- Khởi tạo lại loader nếu loader đã tồn tại (*thêm đoạn code sau vào phương thức onCreate()*)

```

if(getSupportLoaderManager().getLoader(0)!=null){
    getSupportLoaderManager().initLoader(0,null,this);
}

```

- Gọi phương thức **restartLoader()** thay thế cho task **FetchDSBaiTho**

```

getSupportLoaderManager().restartLoader(0, null, this);

```

Phương thức `restartLoader()` được định nghĩa trong lớp `LoaderManager` (*Mỗi Activity có 01 `LoaderManager` quản lý tất cả các loaders được sử dụng trong Activity*)

Phương thức `restartLoader()` có 03 tham số:

- *id*: định danh của 01 loader trong activity.
- *Bundle*: dữ liệu cung cấp cho Loader.
- *Thể hiện của 01 lớp cài đặt `LoaderCallbacks`.*

3. Chạy ứng dụng

- Chạy API **tap-tho**
- Chạy ứng dụng **TapTho**