

# Utilisation du protocole HTTP en Android

## Le protocole HTTP

Pour utiliser le protocole http dans android il y a plusieurs façon de le faire.

Si vous voulez voir la documentation sur la connectivité en Android voila le lien :

<https://developer.android.com/guide/topics/connectivity>

<https://developer.android.com/training/basics/network-ops/connecting>

Il existe un Objet Volley ou Cronet qui permet de faire des requettes HTTP mais je prefere qu l on regarde l'objet HttpURLConnection

<https://developer.android.com/reference/java/net/HttpURLConnection.html>

Cet Objet java qui vient de java.net est aussi utilisable en java swing et fonctionne de la meme facon.

## Permission

Il faudra dans un premier temps ajouter les permissions dans le manifeste pour pouvoir faire des requêtes qui utilise le réseau. Cf documentation.

## Network Configuration

Depuis l'api v9 si l on veut faire des requêtes en http et non en https on doit ajouter une configuration, dans le manifeste on doit ajouter :

```
android:usesCleartextTraffic="true"
```

et on ne pas utiliser localhost comme nom de domaine, il faut utiliser l'adresse ip

ex : <http://192.168.45.32:8080/monsite/maPage>

Rem :

- sous windows, dans une cmd taper la commande ipconfig
- sous linux ctrl+alt+t et taper la commande ifconfig

<https://developer.android.com/studio/run/emulator-networking.html>

## AsyncTask

Lorsque l'on va vouloir faire une requête http depuis l'activité courante on va avoir une exception qui va se déclencher (Network on main thread exception) à vérifier avec les modifications apportées sur les threads en Android.

Dans notre lab pour utiliser l'objet HttpURLConnection il va falloir le combiner avec une AsyncTask pour que cela fonctionne

<https://developer.android.com/reference/android/os/AsyncTask>

Cette classe utilise les génériques et demande de préciser 3 Types.

Le type du paramètre que l'on va envoyer dans la méthode execute(Type ... type) ou doInBackground

Le type de paramètre générer pendant la tâche, utile surtout pour tâche longue, si la tâche est rapide on va mettre Nullable à la place

Le type de Retour de la Tâche. Ici String

Par ex on pourrait avoir

```
class MaTaskPerso extends AsyncTask<String, Nullable, String> {}
```

//Le premier String correspond au type que l'on va récupérer dans la méthode

doInBackground(String ... mesStrings) que l'on doit redéfinir, et dans la méthode execute(String .. s) qui est la méthode pour démarrer la tâche.

// Nullable correspond au type dans la méthode onProgressUpdate(Nullable .. null) dans ce cas je ne l'utilise pas donc je mets null ou le Wrapper de null qui est Nullable

// le dernier String et le type que l'on aura dans la méthode onPostExecute(String ... strings) et correspond aussi au type retourné dans la méthode doInBackground .

On va redéfinir dans doInBackground l'utilisation de HttpURLConnection  
et dans onPostExecute la modification que l'on veut amener au UI de l'activité.

## HttpURLConnection

<https://developer.android.com/reference/java/net/HttpURLConnection.html>

Cet Objet est là pour utiliser le protocole HTTP, on va donc retrouver dessus toutes les méthodes nécessaires pour le manipuler.

Ex :

```
- urlConnection.setMethode("POST");
```

- `urlConnection.getResponseCode()`
- `urlConnection.getInputStream()`
- `urlConnection.getOutputStream()`

Vous pourrez aisément trouver le code pour réaliser la connexion http sur internet.

## Traitement de la réponse

Dans le cadre de ce lab nous allons utiliser le json pour traiter la réponse http

Il existe plusieurs façon de traiter le Json en java, mais celle que j utilise le plus est la librairie de Google soit gson. Il faudra ajouter cette librairie dans les dépendances.

File → ProjetStrucutre

Dans All dependencies cliquer sur le + ajouter une Library Dependency et chercher gson, ça peut prendre jusqu'à une minute mais ensuite il va vous proposer des dépendances correspondant a gson.

Ceci devrait vous être utile : <https://sites.google.com/site/gson/gson-user-guide>

et plus précisément <https://sites.google.com/site/gson/gson-user-guide#TOC-Collections-Examples>

## Déroulement du LAB

Ce lab est fait pour que vous appreniez par vous même.

Toutes les informations ne sont pas dans ce document (mais quand même bcp...).

Comme d'habitude réfléchissez à ce que vous devez faire en premier. Ne pas faire la dernière partie d'un coup.

Allez y par étape.

1 se connecter a l url/connect en ajoutant les paramétrés prenom et password

2 récupérer la réponse et surtout votre guid pour vous connecter à l'api. Vous ne pourrez pas récupérer les datas si vous n'envoyer pas le guid en paramètre (stocker le dans sharepreference)

3 ajouter les boutons des catégories en utilisant la librairie GSON.

4 afficher tous les produits dans l'application au démarrage.

5 essayer de récupérer les images des produits depuis le répertoire assets.

7 afficher les produits par catégorie. 8 télécharger les images depuis le site web et les mettres dans assets



a gauche au démarrage toutes les recettes sont la , si on clique sur une catégorie on voit que les produits de cette catégorie

Les url important :

[https://ip\\_du\\_prof/labAndroid/](https://ip_du_prof/labAndroid/)

[https://ip\\_du\\_prof/labAndroid/connection](https://ip_du_prof/labAndroid/connection)



parametres : prenom et password. Pour tout le monde le mdp est abc123...

[https://ip\\_du\\_prof/labAndroid/categorie](https://ip_du_prof/labAndroid/categorie)

parametre : id

[https://ip\\_du\\_prof/labAndroid/recette](https://ip_du_prof/labAndroid/recette)

parametres : id et idCategorie

[https://ip\\_du\\_prof/labAndroid/result.jsp](https://ip_du_prof/labAndroid/result.jsp)