

Fonctions de callback

Travail dans la classe métier

1° Implémenter une interface de callback dans la classe métier

```
public interface CallbackString{
    public void endGetReponse(List<Valeur> lv);
    public void endErreur(String e);
}
```

Avec autant de méthodes que de types de retour ...

2° Mettre un attribut privé dans la classe métier

```
private CallbackString cb;
```

Cet attribut reçoit le contexte (MainActivity.This)

3° Dans le onResponse ou le OnError on appellera la fonction de callback adéquate en lui passant l'objet à retourner.

```
public void onResponse(JSONArray js)
{
    try {
        . . . . .

        if(cb!=null) cb.endGetReponse(listeVal);
        ....

    } catch (Exception e) {
        . . . . .
    }
}

public void onErrorResponse(VolleyError volleyError)
{
    if(cb!=null) cb.endErreur("erreur connexion");
}
}
```

Travail dans la mainActivity

1° déclarer la fonction de callback

```
public class MainActivity extends AppCompatActivity implements accesBDD.CallbackString{ . . .
```

2° créer les méthodes de callback

```
@Override
public void endGetReponse(List<Valeur> reponse){
    if(reponse !=null){
        . . . . .
    }
}

@Override
public void endErreur(String erreur){
    Toast.makeText(MainActivity.this, erreur, Toast.LENGTH_SHORT).show();
}
}
```