

## Partie 1 : De Microsoft Forms à la Base De Données

### Fiche Employé

votre E-mail est récupéré automatiquement si vous êtes pas content triplé

1. Entrez votre ID \*

Le nombre doit être compris entre 10000 ~ 99999

2. Prenom \*

Entrez votre réponse

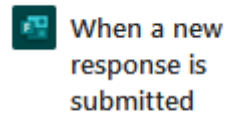
3. Nom \*

Entrez votre réponse

4. Salaire \*

La valeur doit être un nombre

+ Ajouter

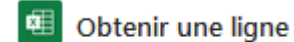


1 ) le déclencheur du flux est la réponse au formulaire que nous avons créé et qui contient quelques questions de base : le nom, le prénom, l'adresse mail et le salaire.

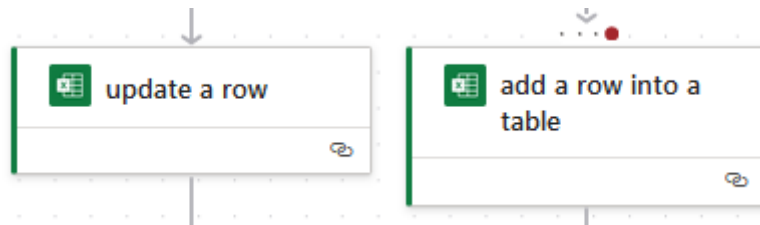


2 ) On récupère les réponses au formulaire

ID	Prenom	Nom	Email	Salaire
1	Jean	Allavena	jean.allavena@gmail.com	3
2	Victor	Asiter	vico.Astier@gmail.com	9000
3	Lucas	Fougère	lucas.fougere@gmail.com	113
4	Nina	Balais	nina.balais@gmail.com	17000
11	Oscaro	Lasticot	stage-info2@vitalis-rtp.fr	66453545
13	nfnfnff	f f f	stage-info2@vitalis-rtp.fr	43
10025	APOOH	iuquetuefedocorc	stage-info2@vitalis-rtp.fr	125
11511	jeCroisEn	Toi	stage-info@vitalis-rtp.fr	20
11886	Attendre	UneMinTrente	stage-info@vitalis-rtp.fr	11889
12345	ouijelà	méweeckclair	stage-info@vitalis-rtp.fr	907
12581	non	APOOOOOH	stage-info2@vitalis-rtp.fr	5630
14987	Oscar	yipiiaiiii	stage-info2@vitalis-rtp.fr	645
17985	jaimewtwo	tatroplarage	stage-info2@vitalis-rtp.fr	987
19853	AmoiAmoi	Amoi	stage-info2@vitalis-rtp.fr	399

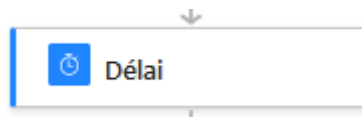


3 ) On parcourt le tableur excel pour voir si l'ID donné dans le formulaire existe ou pas.

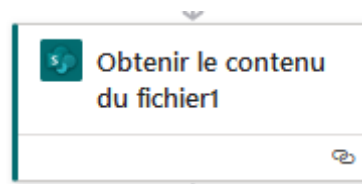


4A ) Si l'ID existe déjà, alors on va modifier la ligne en question avec les informations du formulaire.

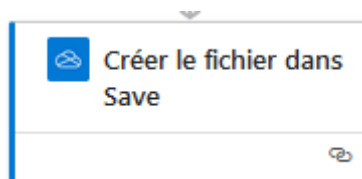
4B ) Sinon, on crée une nouvelle ligne dans le tableur.



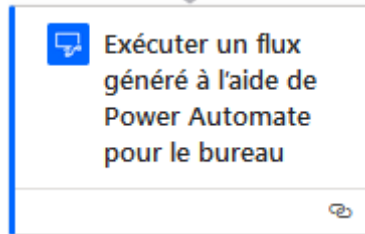
5 ) On ajoute un délai de plusieurs secondes (3 interminables minutes) afin que le dossier s'actualise et que les données s'enregistrent.



6 ) On récupère le contenu du tableur.



7 ) On stock la copie du tableur dans un fichier dans Onedrive.



8 ) Enfin on envoie un signal qui lance un flux [Power Automate Desktop](#) qui est une version différente de Power Automate.

Voici le flux Power Automate Desktop :



On récupère le contenu du fichier stocké dans Onedrive. Les données arrivent sous la forme Byte[XXX].

On utilise ensuite un outil de Power Automate Desktop qui permet de convertir les données binaires en fichier.

On exécute ensuite un script PowerShell qui permet de transformer le fichier qui est actuellement au format .xlsx en .csv

Code PowerShell à exécuter :

```
1 $Excel = New-Object -ComObject Excel.Application
2 $Workbook = $Excel.Workbooks.Open("C:\pgAdmin\Save\TB_CLIENT.xlsx")
3
4 # Sauvegarder en tant que CSV
5 $Workbook.SaveAs("C:\pgAdmin\Save\TB_CLIENT.csv", 6) # 6 est le format CSV
6
7 # Fermer le classeur et quitter Excel
8 $Workbook.Close($false)
9 $Excel.Quit()
10
11 # Libérer les objets COM
12 [System.Runtime.InteropServices.Marshal]::ReleaseComObject($Workbook) | Out-Null
13 [System.Runtime.InteropServices.Marshal]::ReleaseComObject($Excel) | Out-Null
14
```

Enfin on exécute un autre script PowerShell qui va se déplacer dans un répertoire qui contient un script .bat qui appelle un script SQL pour insérer les données du fichier .csv dans la base de données.

Code PowerShell à exécuter :

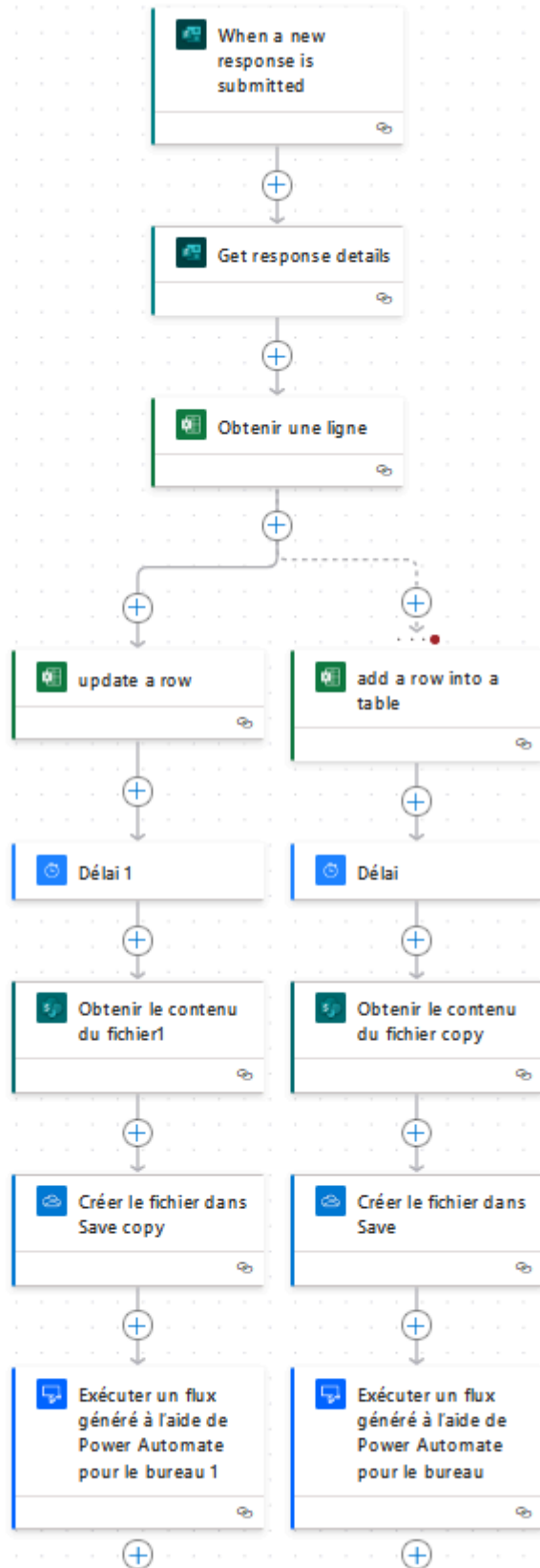
```
1 cd C:\pgAdmin\Save\
2
3 Start-Process -FilePath ./script.bat
```

script.bat

```
set PGPASSWORD=admin1
psql -h localhost -U postgres -d employe -f creationTable.sql
set PGPASSWORD=
```

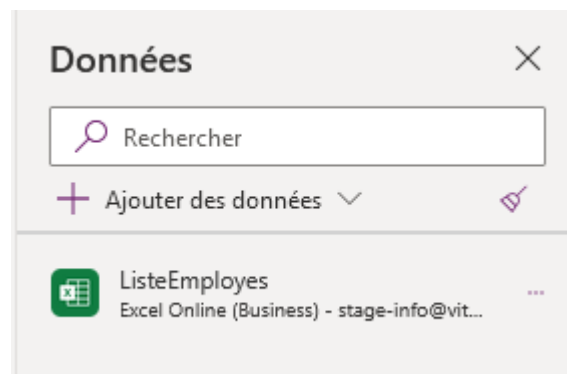
creationTable.sql

```
COPY "miaou" ("ID", "Prenom", "Nom", "Email", "Salaire")
FROM 'C:\pgAdmin\Save\TB_EMPLOYE.csv'
DELIMITER ';' CSV HEADER;
```



## Partie 2 : Power Apps et la BDD dans les deux sens

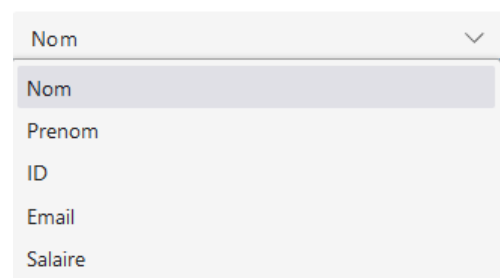
On crée un tableur excel qui servira de base pour l'application Power Apps.  
On crée l'application dans Power Apps ([voir Fiche de procédure Power Apps](#))



Après avoir créé l'application, on la modifie et on y ajoute différents éléments.  
On obtient le résultat suivant :

Email	ID
nina.balais@gmail.com	40000
Nom	Prenom
Balais	Nina
Salaire	
17001 €	

**1)** La barre de recherche permet de filtrer les résultats de la galerie ci-dessous  
Le menu déroulant permet de choisir sur quel attribut le filtre va s'appliquer





2 ) La galerie contient tous les éléments du tableur (seuls le Mail, l'ID et le Nom sont affichés ). On peut scroller pour voir tous les éléments ou en sélectionner un pour plus d'informations.

Le code pour le filtre est dans cette partie :

```
Switch(FilterMenu.Selected.Value;  
    "ID";Filter([@ListeEmployes];Value(ID) = Value(SearchInput1.Text));  
    "Nom";Filter([@ListeEmployes];StartsWith(Nom; SearchInput1.Text));  
    "Prenom";Filter([@ListeEmployes];StartsWith(Prenom; SearchInput1.Text));  
    "Email";Filter([@ListeEmployes];StartsWith(Email; SearchInput1.Text));  
    "Salaire";Filter([@ListeEmployes];Value(Salaire) = Value(SearchInput1.Text))  
)
```

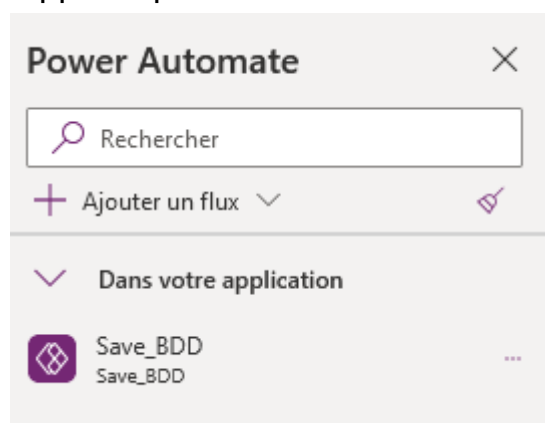
3 ) Lorsque l'on sélectionne un élément de la galerie, le reste des informations s'affiche dans le corps de la page

4 ) On a la possibilité d'ajouter, de supprimer ou de modifier les éléments. Le visuel change lorsque l'on clique sur les icônes

nina.balais@gmail.com ✓ ✕ Edit

Email	ID
<input type="text" value="nina.balais@gmail.com"/>	<input type="text" value="40000"/>
Nom	Prenom
<input type="text" value="Balais"/>	<input type="text" value="Nina"/>
Salaire	
<input type="text" value="17001 €"/>	

5 ) Avec Power Apps on peut lancer des flux Power Automate.



La disquette permet de déclencher un flux Power Automate  
(plus ou moins le même que celui de la partie une ci-dessus)

