

# Curl

Voici un script bash qui combine plusieurs exemples de curl :

```
#!/bin/bash

# Requête GET de base
echo "Requête GET de base:"
curl http://example.com

# GET avec des en-têtes personnalisés
echo "\nRequête GET avec des en-têtes personnalisés:"
curl -H "User-Agent: MyUserAgent/1.0" http://example.com

# Requête POST
echo "\nRequête POST:"
curl -X POST -d "key1=value1&key2=value2" http://example.com/post

# POST avec des données JSON
echo "\nPOST avec des données JSON:"
curl -X POST -H "Content-Type: application/json" -d '{"key1": "value1", "key2": "value2"}' http://example.com/post

# Utilisation d'un proxy
echo "\nUtilisation d'un proxy:"
curl -x http://proxy.example.com:8080 http://example.com

# Authentification
echo "\nAuthentification de base:"
curl -u username:password http://example.com/auth

# Téléchargement d'un fichier
echo "\nTéléchargement d'un fichier:"
curl -O http://example.com/file.zip

# Suivre les redirections
echo "\nSuivre les redirections:"
curl -L http://example.com/redirect

# Enregistrer la sortie dans un fichier
echo "\nEnregistrer la sortie dans un fichier:"
```

```
curl http://example.com -o output.txt
```

```
# Méthode HTTP personnalisée
```

```
echo "\nMéthode HTTP personnalisée (DELETE):"
```

```
curl -X DELETE http://example.com/delete/resource
```

```
# Sortie verbose
```

```
echo "\nSortie verbose:"
```

```
curl -v http://example.com
```

```
# Délai de connexion
```

```
echo "\nDélai de connexion:"
```

```
curl --connect-timeout 5 http://example.com
```

```
echo "\nExécution du script terminée."
```

Enregistrez ce script avec une extension `.sh`, comme `curl_examples.sh`. Assurez-vous de le rendre exécutable si vous êtes dans un environnement de type Unix ou si vous l'exécutez dans Windows Subsystem for Linux (WSL) :

```
chmod +x curl_examples.sh
```

Ensuite, vous pouvez l'exécuter :

```
./curl_examples.sh
```

**Notes pour Windows :** - Si vous exécutez ceci dans Windows Cmd ou PowerShell, vous devrez utiliser `bash` ou `wsl` si vous avez WSL installé, ou vous assurer que `curl` est dans votre PATH pour une exécution directe : - `bash curl_examples.sh` OU `wsl ./curl_examples.sh` - Notez que ce script utilise la syntaxe `bash`, donc il n'est pas nativement compatible avec l'invite de commandes Windows ou PowerShell sauf s'il est exécuté via WSL ou un environnement `bash`. Cependant, les commandes `curl` elles-mêmes sont compatibles avec la version Windows de `curl`.