

## TD1 : routes, json

L'objectif du TD est de mettre en place et commencer à maîtriser les bases de votre framework d'application et de votre environnement de développement.

### 1. Programmation : routes, production de json

**Contexte** : on réalise une API pour un service de prise de commandes pour une application de vente de sandwiches. Une commande est composée de plusieurs items, chaque item contenant simplement un label, le tarif et la quantité commandée, ainsi qu'une référence vers un élément dans un catalogue.

Programmer les 2 routes suivantes, qui **doivent retourner des données en json structurées comme indiqué dans les exemples**. La réponse retournée doit également contenir le header `Content-Type` : avec la valeur adéquate (`application/json; charset=utf-8`).

On n'utilise pas de base de données : les données sont (pour l'instant) stockées dans des variables.

Pour tester et interagir avec votre micro api, il est nécessaire d'utiliser une application dédiée comme Postman (<https://www.getpostman.com/>) ou Insomnia (<https://insomnia.rest/>).

a) la route `/commandes` : retourne la liste des commandes déjà créées sous la forme d'un objet json. On utilisera une liste de 3 ou 4 commandes, chaque commande étant décrite par un `Id`, un `mail_client`, une `date_commande`, un montant. Le json produit **doit avoir la forme suivante** :

```
{
  "type": "collection",
  "count": 3,
  "commandes": [
    {
      "id": "AuTR4-65ZTY",
      "mail_client": "jan.neymar@yaboo.fr",
      "date_commande": "2022-01-05 12:00:23",
      "montant": 25.95
    },
    {
      "id": "657GT-I8G443",
      "mail_client": "jan.neplin@gmal.fr",
      "date_commande": "2022-01-06 16:05:47",
      "montant": 42.95
    },
    {
      "id": "K9J67-4D6F5",
      "mail_client": "claudfrancois@grorange.fr",
      "date_commande": "2022-01-07 17:36:45",
      "montant": 14.95
    }
  ]
}
```

b) la route `/commandes/{id}` : retourne la description complète de la commande indiquée, sous la forme d'un objet json :

- `/commandes/K9J67-4D6F5` : retourne la description de la commande K9J67-4D6F5

Le json produit doit avoir la forme suivante :

```
{
  "type": "resource",
  "commande": {
    "id": "K9J67-4D6F5",
    "mail_client": "claude.francois@grorange.fr",
    "nom_client": "claude francois",
    "date_commande": "2022-01-07 17:36:45",
    "date_livraison": "2022-01-08 12:30",
    "montant": 14.95
  }
}
```