



Exercise



httpClient

Récupérer le client

```
$client = \Drupal::httpClient();
```

Effectuer une requête

```
$client->request('GET', 'http://demo.ckan.org/api/3/action/package_list');
```

Requête GET + réponse

```
$request = $client->get('http://demo.ckan.org/api/3/action/package_list');  
$response = json_decode($request->getBody());
```

POST + réponse

```
$client = \Drupal::httpClient();  
$request = $client->post('http://demo.ckan.org/api/3/action/group_list', [  
    'json' => [  
        'id' => 'data-explorer'  
    ]  
]);  
$response = json_decode($request->getBody());
```



Utilisation d'une API externe

<https://swapi.co/>

Try it now!

<https://swapi.co/api/> [people/1/](https://swapi.co/api/people/1/)

request

Need a hint? try [people/1/](https://swapi.co/api/people/1/) or [planets/3/](https://swapi.co/api/planets/3/) or [starships/9/](https://swapi.co/api/starships/9/)

Result:

```
{
  "name": "Luke Skywalker",
  "height": "172",
  "mass": "77",
  "hair_color": "blond",
  "skin_color": "fair",
  "eye_color": "blue",
  "birth_year": "198BY",
  "gender": "male",
  "homeworld": "https://swapi.co/api/planets/1/",
  "films": [
    "https://swapi.co/api/films/2/",
    "https://swapi.co/api/films/6/",
    "https://swapi.co/api/films/3/",
    "https://swapi.co/api/films/1/",
    "https://swapi.co/api/films/7/"
  ],
  "species": [
```



- Création de 3 vocabulaires (climat, terrai, species)
- Création de 3 entités Planet, Starship, People
- Création d'un service : StarWarsService (création des entités via l'API)
- Création d'un formulaire d'ajout de personnage (création des entités via le service)
- Création d'une page de listing des personnages (Controller + hook_theme + template)
- Création d'un Block affichant un personnage (configurable sur le formulaire du block)



Etape 1

Créer les vocabulaires suivant :

- Climat (climat)
- Terrain (terrain)
- Species (species)

Afficher le poids

NOM DU VOCABULAIRE	DESCRIPTION	OPERATIONS
 Climat		Lister les termes 
 Species		Lister les termes 
 Terrain		Lister les termes 

Etape 2



Créer l'entité Planet (starwars_planet) à l'aide de la commande *drupal generate:entity:content*

Ajouter ensuite les champs suivants à l'entité :

- Terrain – field_terrain (référence à des termes de taxonomie Terrain)
- Climat – field_climat (référence à des termes de taxonomie Climat)
- Population – field_population (Texte brut)

Une planète a un nom, différents types de terrain, différents climats et une population.

Etape 2



Créer l'entité Starship (starwars_starship) à l'aide de la commande *drupal generate:entity:content*

Ajouter ensuite le champ suivant à l'entité :

- Model – field_model (Texte brut)

Un vaisseau à un nom et un modèle.

Etape 2



Créer l'entité People (starwars_people) à l'aide de la commande *drupal generate:entity:content*

Ajouter ensuite le champ suivant à l'entité :

- Gender – field_gender (Texte brut)
- Species – field_species (référence à un terme de taxonomie Species)
- Homeworld – field_homeworld (référence à une entité Planet)
- Starships – field_starships (référence à des entités Starship)

Un personnage a un sexe, une espèce, une planète d'origine et des vaisseaux.

Etape 3

- Créer un module starwars
- Créer une interface pour notre service voir exemple ->

```
<?php

/**
 * File : starwars\src\Service\StarwarsInterface.php
 */

namespace Drupal\starwars\Service;

/**
 * Interface StarwarsInterface.
 */
interface StarwarsInterface {

    /**
     * Add a StarWars Character
     *
     * @param integer $id The StarWars People Id
     *
     * @return mixed
     */
    public function addPeople($id);

    /**
     * Add a StarWars Planet
     *
     * @param string $uri The Swapi URI
     *
     * @return string $id An Entity Planet Id
     */
    public function addPlanet($uri);

    /**
     * Get a Species name from the Swapi URI
     *
     * @param string $uri The Swapi URI
     *
     * @return string The species name
     */
    public function getSpecies($uri);

    /**
     * Add a StarWars StarShip
     *
     * @param string $uri The Swapi URI
     *
     * @return string $id An Entity Starship Id
     */
    public function addStarship($uri);
}
```





Etape 3

- Créer le service Starwars implémentant l'interface StarwarsInterface
- Ajouter le service au fichier starwars.services.yml

```
<?php

namespace Drupal\starwars\Service;

use Drupal\Core\Config\ConfigFactory;
use Drupal\starwars\Entity\People;
use Drupal\starwars\Entity\Planet;
use Drupal\taxonomy\Entity\Term;

/**
 * Class Starwars.
 */
class Starwars implements StarwarsInterface {

    protected $httpClient;

    /**
     * Constructs a new Starwars object.
     */
    public function __construct() {
        $this->httpClient = \Drupal::httpClient();
    }

}
```



Etape 4

- Créer un formulaire (starwars/formulaire) d'ajout de personnage via un champ id.
- Injecter le service StarWars dans la classe du formulaire
- Code de la fonction submit :

```
$values = $form_state->getValues();  
$id = $values['id_people'];  
$this->starwarsDefault->addPeople($id);
```

- Le terrain lakes a été ajouté
- La planète Endor a été ajoutée
- L'espèce Ewok a été ajoutée
- Le personnage Wicket Systri Warrick a été ajouté

Ajouter un personnage

ID PEOPLE

Soumettre



Etape 5

- Remplissez les différentes fonctions du service StarWars afin d'ajouter en cascade les différentes entités.
- Exemples de code :

```
// Vérifier si personnage existe déjà
$ids = \Drupal::entityQuery('starwars_people')
    ->condition('name', $people->name)
    ->execute();
// Si le personnage n'existe pas
if (empty($ids)) {
```

```
// Créer un term Terrain
$term = Term::create(
[
    'vid' => 'terrain',
    'name' => $terrain,
]
);
```

```
// Vérifier si un terme d'une espece a déjà été ajouté
$specieTerm = \Drupal::entityQuery('taxonomy_term')
    ->condition('vid', 'terrain')
    ->condition('name', $specie)
    ->execute();

$specieTerm = current($specieTerm);

if (empty($specieTerm)) {
```

```
// Ajouter un personnage
$newPeople = People::create([
    'name' => $people->name,
    'field_homeworld' => $planet, // id de l'entité planet
    'field_species' => $specieTerm, // id ou tableau d'ids exemple [15,45,20] du
    terme species
]);
$newPeople->save();
```

```
// Parser une réponse
$request = $this->httpClient->get($uri);
$planet = json_decode($request->getBody());
$nom_planete = $planet->name;
```

- Créer une page Liste des personnages (listant tous les personnages enregistrés) ->
 - Controller
 - Route
 - hook_theme
 - template twig

```
// Récupérer les entités référencées d'un champ
$planet = $entity->get('field_homeworld')->referencedEntities();
```

Liste des personnages

Personnage	Planete	Vaisseaux
Nom : Luke Skywalker	Nom : Tatooine	Imperial Speeder Bike (74-Z speeder bike)
Espece : Human	Terrains : desert	Snowspeeder (t-47 aircspeeder)
Sexe : male	Climat : arid	X-wing (T-65 X-wing)
	Population : 200000	

Personnage	Planete	Vaisseaux
Nom : Luke Skywalker	Nom : Tatooine	Imperial Speeder Bike (74-Z speeder bike)
Espece : Human	Terrains : desert	Snowspeeder (t-47 aircspeeder)
Sexe : male	Climat : arid	X-wing (T-65 X-wing)
	Population : 200000	

Etape 7



Créer un block avec un formulaire permettant de choisir l'entité à afficher dans celui-ci.

- Création d'un block avec form
- hook_theme
- template twig

```
// champ d'autocomplétion personnage
$form['starwars_people'] = array(
  '#type' => 'entity_autocomplete',
  '#title' => 'People',
  '#target_type' => 'starwars_people',
);
```