###### وزارة التعليم العالي والبحث العلمي

|  |  |  |
| --- | --- | --- |
| **جامعة باجي مختار – عنابـــــــــــــــة** |  | **Université Badji Mokhtar - Annaba**  **Badji Mokhtar – Annaba University** |
| **Faculté : Technologie** | | |
| **Département : Informatique** | | |
| **Domaine : Math et Informatique** | | |
| **Filière : M2 SID** | | |
|  | | |

**Web mining: Extraction de Données à partir d’un Site**

**Web**

**Présenté par :**

* + - Deneche Mohamed Islem
    - Horchi Abir Selma

**Mme.Lagrini Université** : UBMA

Année Universitaire : 2025/2026

## Site analysé : [www.tct.pro](http://www.tct.pro) (catalogue de véhicules)

## 1. Complétude des Données

### Méthode 1 : Extension de navigateur (Web Scraper)

**Données extraites : 11 champs**

* web-scraper-order
* web-scraper-start-url
* image-src
* Nom du véhicule (title)
* Année (year)
* Prix (price)
* Kilométrage (mileage)
* Transmission
* Carburant (fuel)
* Disponibilité (status)
* URL de départ
* **Manquant** : galerie d'images

**Taux de complétude : (11/11 champs)**

### Méthode 2 : API publique

**Données extraites : 11 champs**

* ID
* Nom (name)
* Année (year)
* Prix (price)
* Kilométrage (mileage)
* Transmission
* Carburant (fuel)
* Image principale
* Galerie d'images (gallery)
* Badge de disponibilité
* État de disponibilité (isAvailable)

**Taux de complétude : (11/11 champs)**

### Méthode 3 : Bibliothèques Python

**Données extraites : 9 champs**

* ID
* Nom (name)
* Année (year)
* Prix (price)
* Kilométrage (mileage)
* Transmission
* Carburant (fuel)
* Badge
* isAvailable
* **Manquant** : Image, galerie

**Taux de complétude : (9/11 champs)**

### Méthode 4 : Plateforme sans code

**Données extraites : 8 champs**

* Nom (Name)
* Image
* Badge
* Année (Year)
* Prix (Price)
* Kilométrage (Mileage)
* Transmission
* Carburant (Fuel)
* **Manquant** : ID, galerie, isAvailable

**Taux de complétude : (8/11 champs)**

## 2. Facilité d'Utilisation

### 1er : Plateforme sans code

* **Difficulté : (Très facile)**
* Interface graphique intuitive
* Aucune programmation requise
* Configuration automatique pour l’extraction des données
* Idéal pour les débutants

### 2ème : API publique

* **Difficulté : (Facile)**
* Une seule requête HTTP suffit
* Documentation claire (supposée)
* Pas besoin d'analyser le HTML
* Requiert des connaissances basiques en programmation

### 3ème : Extension de navigateur

* **Difficulté : (Moyen)**
* Configuration des sélecteurs CSS nécessaire
* Interface semi-technique
* Peut nécessiter plusieurs essais pour les bons sélecteurs
* Documentation à consulter

### 4ème : Bibliothèques Python

* **Difficulté : (Difficile)**
* Écriture de code complète
* Gestion des erreurs
* Parsing HTML/JSON manuel
* Installation de dépendances (requests, BeautifulSoup, etc.)

## 3. Temps d'Extraction

### Pour 20 véhicules :

| **Méthode** |  |  | **Détails** |
| --- | --- | --- | --- |
| **API publique** |  |  | Une seule requête pour tout récupérer |
| **Bibliothèques Python** |  |  | Dépend du code (requêtes multiples possibles) |
| **Extension navigateur** |  |  | Chargement du navigateur + extraction |
| **Plateforme sans code** |  |  | Configuration par defaut + extraction + export |

## 4. Précision des Résultats

### API publique :

* Données structurées et fiables
* Format CSV standardisé
* Aucune erreur de formatage
* Toutes les données présentes et correctes

### Bibliothèques Python : Très bonne

* Données correctes et bien formatées
* Possible extraction incomplète de la galerie d'images
* Dépend de la qualité du code écrit

### Extension navigateur : Bonne

* Données correctes mais incomplètes
* Risque d'erreurs si le site change sa structure
* Formatage parfois inconsistant

### Plateforme sans code : Bonne

* Données visuelles bien extraites
* URL d'images complexes .
* Manque de données techniques (ID, isAvailable)

## 5. Problèmes Rencontrés

### Extension de navigateur

* Présence des données qui n’ont pas de sens
* L’existence d’une autre format d’ID
* Dépendance à la structure HTML (fragile aux changements)
* Configuration des sélecteurs CSS parfois laborieuse

### API publique

* Aucun problème majeur
* Nécessite de découvrir l'endpoint API (inspection réseau)
* Possible limitation de taux (rate limiting)

### Bibliothèques Python

* Galerie d'images non extraite (non accessible dans le HTML initial)
* Gestion des erreurs réseau nécessaire
* Parsing HTML complexe pour certaines données
* Possible besoin de Selenium pour du contenu dynamique

### Plateforme sans code

* Limitations sur les données complexes (pas d'ID)
* URLs d'images avec paramètres lourds
* Moins de contrôle sur le processus d'extraction
* Coût potentiel pour versions avancées

## 6. Avantages et Inconvénients

### Extension de navigateur

**Avantages :**

* Interface visuelle intuitive
* Pas besoin de coder
* Visualisation en temps réel
* Export facile en CSV
* Gratuit

**Inconvénients :**

* Lent pour de gros volumes
* Fragile aux changements du site
* Limité aux données visibles

API publique

**Avantages :**

* **Meilleure complétude** (100% des données)
* Très rapide et efficace
* Données structurées et fiables
* Facile à automatiser
* Respectueux du serveur

**Inconvénients :**

* Nécessite de trouver l'API
* Peut être protégée ou limitée
* Requiert des compétences en programmation
* Risque de changement sans préavis

### Bibliothèques Python

**Avantages :**

* Grande flexibilité et contrôle
* Automatisation complète possible
* Gratuit et open source
* Communauté active

**Inconvénients :**

* Courbe d'apprentissage élevée
* Code à maintenir
* Galerie d'images difficile à extraire
* Peut nécessiter Selenium (plus lourd)

### Plateforme sans code

**Avantages :**

* Aucune compétence technique requise
* Interface drag-and-drop
* Support client disponible
* Exports multiples formats

**Inconvénients :**

* Données incomplètes (pas d'ID, pas isAvailable)
* URLs d'images complexes
* Moins de contrôle
* Coût potentiel
* Limité pour des cas avancés

## Classement Global

### Par complétude des données :

1. **API publique** 11/11 champs
2. **Bibliothèques Python** 9/11 champs
3. **Extension navigateur** 11/11 champs
4. **Plateforme sans code** 8/11 champs

### Par facilité d'utilisation :

1. **Plateforme sans code** - Aucune compétence requise
2. **API publique** - Simple requête HTTP
3. **Extension navigateur** - Configuration visuelle
4. **Bibliothèques Python** - Code complet nécessaire

### Par rapidité d'extraction :

1. **API publique** (très rapide)
2. **Bibliothèques Python** (rapide)
3. **Extension navigateur** (peu de temps)
4. **Plateforme sans code** (peu de temps)

## Recommandations

**Pour ce projet spécifique (**[**www.tct.pro**](http://www.tct.pro)**) :**

* **Meilleur choix global : API publique**
  + Données complètes, rapide, fiable
  + Idéal pour un usage récurrent
* **Pour débutants : Plateforme sans code**
  + Suffisant pour des besoins basiques
  + Aucune barrière technique
* **Pour apprentissage : Bibliothèques Python**
  + Développe des compétences en programmation
  + Flexible et réutilisable
* **Extension navigateur : Usage ponctuel uniquement**
  + Bon pour des tests rapides
  + Trop limité pour un usage sérieux

## Conclusion

L'**API publique** se distingue clairement comme la méthode optimale pour ce site, offrant la complétude totale des données (100%), la rapidité d'extraction la plus élevée, et une fiabilité maximale. Cependant, le choix final dépend du contexte : un utilisateur sans compétences techniques préférera la plateforme sans code, tandis qu'un développeur optera pour l'API ou les bibliothèques Python pour plus de flexibilité et d'automatisation.