



## 4 - Bonnes pratiques et organisation du code

### Structure du code JavaScript

#### Organisation recommandée

Pour index.html :

Plain Text | ...

```
1. Constantes de configuration (URL de l'API) 2. Fonction de récupération des données 3. Fonction de création d'une carte 4. Fonction de rendu de toutes les cartes 5. Fonction d'initialisation (DOMContentLoaded) 6. Gestion des erreurs
```

Pour car.html :

Plain Text | ...

1. Constantes de configuration 2.  
Fonction d'extraction de l'ID 3.  
Fonction de récupération des données  
4. Fonctions de rendu (specs,  
highlights, images) 5. Fonction  
d'initialisation 6. Gestion des  
erreurs

## Principes de développement

### DRY (Don't Repeat Yourself)

- Créez des fonctions réutilisables
- Évitez la duplication de code
- Factorisez la logique commune

### Séparation des préoccupations

- Séparez la logique de récupération des données du rendu
- Une fonction = une responsabilité claire
- Facilite le debugging et la maintenance

### Nommage clair et cohérent

- Utilisez des noms de variables descriptifs
- Convention camelCase pour JavaScript

- Noms de fonctions verbeux (ex:  
`fetchAllCars` , `renderCarCard` )