* La fonction ***\_\_init\_\_*** lit un fichier XML de tâche planifiée Windows et en extrait toutes les informations importantes : qui l’a créée, quand elle s’exécute, dans quelles conditions, avec quels privilèges, et ce qu’elle exécute. Elle transforme ce fichier en un **objet Python structuré et prêt à l’usage**, ce qui facilite l’analyse, l’affichage ou la modification automatique de tâches planifiées.

|  |  |  |
| --- | --- | --- |
| **1. Informations générales (RegistrationInfo)**  uri  security\_descriptor  source  date  author  version  description  documentation | **2. Informations sur l’utilisateur (Principal)**  principal\_id  user\_id  logon\_type  group\_id  display\_name  run\_level  process\_token\_sid\_type  required\_privileges (liste de privilèges) | **3. Paramètres de configuration (Settings)**  allow\_start\_on\_demand  disallow\_start\_on\_batteries  stop\_on\_batteries  allow\_hard\_terminate  start\_when\_available  network\_profile\_name  run\_only\_on\_network  wake\_to\_run  enabled  hidden  delete\_expired  execution\_time\_limit  run\_only\_idle  unified\_scheduling\_engine  disallow\_start\_on\_remote\_app\_session  multiple\_instances\_policy  priority  idle\_duration  idle\_wait\_timeout  idle\_stop\_on\_idle\_end  idle\_restart\_on\_idle  network\_name  network\_id  restart\_on\_fail\_interval  restart\_on\_fail\_count |
| **4. Données brutes**  data (contenu brut de la balise <Data>) | **5. Déclencheurs**  triggers (liste des conditions de déclenchement de la tâche) | **6. Actions**  actions (liste des actions à exécuter, ex : lancement de programme) |

* La fonction ***\_strip\_namespace* nettoie le XML** en enlevant les informations de namespace, pour que le reste du programme puisse lire les balises **plus facilement et sans erreurs**.  
  C’est une étape de préparation indispensable avant de traiter le fichier XML.
* La fonction \_***get\_element\_data*** sert à **récupérer le texte contenu dans une balise XML** à partir d’un **chemin donné** (path), par exemple "RegistrationInfo/Author".
* La fonction ***\_get\_list\_element\_data*** permet de lire plusieurs balises similaires dans un XML (comme une liste de privilèges) et d’en extraire tous les textes dans une liste Python.

Elle est utile quand plusieurs balises de même nom sont attendues à un même endroit.

* La fonction ***\_get\_list\_tag\_data*** retourne la **liste des noms de balises XML** présentes à un endroit précis.  
  Elle est utile si on veut savoir **quels types d'éléments** sont présents, sans lire leur contenu.
* La fonction ***\_get\_dict\_data*** permet d’extraire des blocs de données XML similaires (comme des déclencheurs ou actions) et de les convertir en dictionnaires Python structurés, regroupés dans une liste.

C’est très utile pour manipuler facilement des ensembles de données complexes depuis un fichier XML.

* La fonction ***\_get\_raw\_xml*** permet de récupérer une **section XML complète sous forme de texte brut**, utile si on veut **garder la structure XML d’origine** ou l’enregistrer telle quelle, sans l’analyser ni la modifier.
* La fonction ***\_get\_triggers*** analyse tous les déclencheurs définis dans une tâche planifiée (dans un fichier XML) et extrait pour chacun son type, ses dates, son état, ses répétitions, ses délais, et d’autres paramètres spécifiques selon le type.

Elle transforme tout cela en une liste de dictionnaires Python faciles à exploiter ensuite dans le code.

* La fonction ***\_get\_schedule\_by\_week*** permet de lire combien de semaines doivent s'écouler entre deux exécutions, et quels **jours de la semaine** sont programmés.  
  Elle renvoie ces données sous forme d’un **dictionnaire** facile à utiliser en Python.
* La fonction ***\_get\_schedule\_by\_month\_day\_of\_week*** lit une configuration avancée de calendrier dans un fichier XML : quelles semaines, quels jours, et quels mois doivent être utilisés pour exécuter une tâche.

Elle retourne un dictionnaire Python structuré contenant ces 3 catégories de données, pour une utilisation facile dans le reste du programme.

* La fonction ***\_get\_actions*** lit toutes les actions associées à une tâche planifiée dans le fichier XML.  
  Elle crée pour chaque action un dictionnaire avec son type et toutes ses propriétés spécifiques (exécuter un programme, envoyer un email, afficher un message, etc.).  
  Le résultat est une liste structurée et facile à manipuler pour exécuter ou analyser ces actions.
* La méthode ***parse*** rassemble toutes les informations extraites d’une tâche XML dans un dictionnaire unique, prêt à être utilisé pour analyser, manipuler ou afficher la configuration complète de cette tâche.

C’est une sorte de résumé final, rendant les données faciles à accéder en un seul objet Python.