1. Довольно странно постоянно сохранять html файл, потом открывать его и парсить данные из него (по сути просто лишние движения). Правильно будет просто использовать responce.text и доставать информацию оттуда (но возможно у заказчика было требования сохранить html файлы)

2. Хардкод пагинации может привести к тому что в будущем не все стартапы будут пасриться (на сайт банально добавят новые стартапы и пагинаций станет больше). Правильно будет проверять ответ на каждой итерации цикла, и если responce.status\_code == 404 то выходим из цикла

3. Это уже ошибка в стиле написания кода. Абсолютно весь код находится в одной функции. Получения запроса с основного сайта и распарсивание карточек стартапов должно быть отдельно (для сумасшедших можно даже на 2 функции разбить). Парсинг данных со страницы стартапа так же должно производится отдельной функцией. Сохранения json, так же должно быть отдельной функцией.

4. Это скорее совет чем ошибка. На сайте не так много карточек, но если учесть что в будующем их вероятно станет больше (запросов станет больше), то код может выполняться несколько десятков минут. Если вы пишите парсер и либо он уже делает много запросов, либо есть вероятность что в будующем он будет делать много запросов, то просто необходимо использовать aiohttp (или любую другую асинхронную библиотеку). Если парсер одноразовый, это можно упустить.

5.Было бы правильно вывести базовый URL в отдельную переменную, а не копировать его к каждой относительной ссылке.

6. Считается неправильным отлавливать все исключения в блоке try except, так как если пользователь нажмёт CTRL+C, то это не прекратит работу программы, а выкинет исключение. Если вам нужно поймать конкретное исключение, укажите его в блоке except, если нужно отловить все (за исключением нажатия CTRL+C), то просто внутри блока except выполняйте проверку.

7. Заголовки лучше прописывать вне функции как константу (в ином случаи вы либо используете одну функцию на всю программу (что неправильно) либо копипастите headers в тело каждой функции)

8. Что бы не заниматься лишними телодвижениями, можно использовать XPATH для указания необходимого блока.

Думаю это всё к чему смог придраться. Отличный урок особенно для новичков. На парсерах можно заработать первый деньги питон разработчику, так как офферов достаточно много. Дам совет для тех кто парсит какой то сложный сайт. Изучите траффик, который идёт от сайта. Иногда можно найти запрос в котором все данные поступают в одном json ответе и вам не нужно долбить сайт сотнями запросов. Если вам удалось найти такой запрос, считайте что вы выполнили заказ на фрилансе за 5 минут, так как вам остаётся лишь достать данные из json и сохранить туда, куда укажет заказчик.