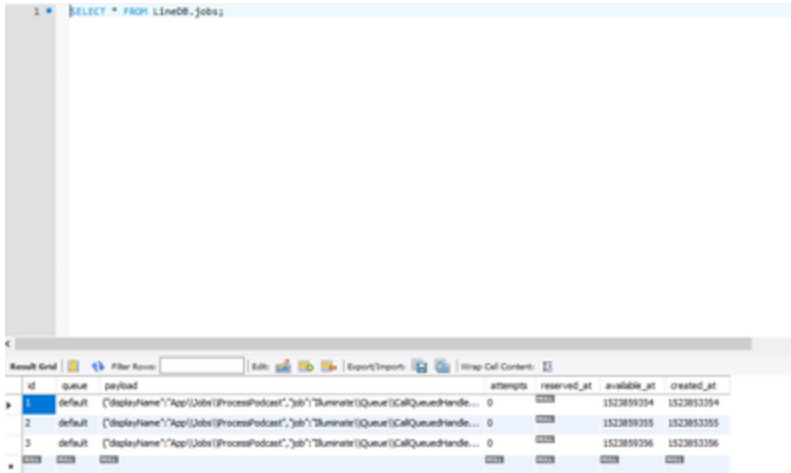


Laravel Job Queues學習日誌

建立Laravel 的Job Queues有以下前置作業

1. 建立Job table 請執行以下指令
 - a. php artisan queue:table
 - b. php artisan migrate
2. 建立fail table 請執行以下指令 (fail table 是用來存執行queue內的job失敗的工作清單)
 - a. php artisan queue:failed-table
 - b. php artisan migrate
3. 在.env檔案內將QUEUE_DRIVER=sync 改成QUEUE_DRIVER=database， 這個動作是將預設的queue driver 改成database

以上前置作業主要是建立job queue的表格 一個是工作清單的表格 一個是失敗的工作清單的表格，最後設定使用的driver是database 不然預設是sync，這樣他不會把job寫入DB的job 表格內，下圖是有寫入的狀，sync應該是直接處理。



id	queue	payload	attempts	reserved_at	available_at	created_at
1	default	("displayName":"App\Jobs\ProcessPodcast","job":"Illuminate\Queue\CallQueueHandle...")	0		1523859394	1523853394
2	default	("displayName":"App\Jobs\ProcessPodcast","job":"Illuminate\Queue\CallQueueHandle...")	0		1523859395	1523853395
3	default	("displayName":"App\Jobs\ProcessPodcast","job":"Illuminate\Queue\CallQueueHandle...")	0		1523859396	1523853396

如果要讓Job Queue內的工作被執行 必須在terminal 下執行 php artisan queue:work (或者 執行 php artisan queue:listen) 去listen 說那些工作有進來了，那些工作可以被執行了。

如果執行工作失敗會有類似以下的畫面

```
root@localhost:/var/www/html/85c# php artisan queue:work
[2018-04-16 13:15:29] Processing: App\Jobs\ProcessPodcast
[2018-04-16 13:15:29] Processing: App\Jobs\ProcessPodcast
[2018-04-16 13:15:29] Processing: App\Jobs\ProcessPodcast
[2018-04-16 13:15:29] Failed: App\Jobs\ProcessPodcast
[2018-04-16 13:15:29] Processing: App\Jobs\ProcessPodcast
[2018-04-16 13:15:29] Failed: App\Jobs\ProcessPodcast
[2018-04-16 13:16:05] Processing: App\Jobs\ProcessPodcast
[2018-04-16 13:16:05] Processing: App\Jobs\ProcessPodcast
[2018-04-16 13:16:05] Failed: App\Jobs\ProcessPodcast
```

在資料庫的failed_jobs的表格可以看到失敗的資料

1 * `SELECT * FROM LineDB.failed_jobs;`

id	connection	queue	payload	exception	failed_at
3	database	default	("deployname":"App\Jobs\ProcessPodcast","job":"Summate(SQ...	PDOException: SQLSTATE[42S22]: Column not found: 1054 Unknown col...	2018-04-16 13:15:29
2	database	default	("deployname":"App\Jobs\ProcessPodcast","job":"Summate(SQ...	PDOException: SQLSTATE[42S22]: Column not found: 1054 Unknown col...	2018-04-16 13:15:29
1	database	default	("deployname":"App\Jobs\ProcessPodcast","job":"Summate(SQ...	PDOException: SQLSTATE[42S22]: Column not found: 1054 Unknown col...	2018-04-16 13:16:05

如果在terminal下執行 `php artisan queue:failed` 可以看到目前資料庫內存的資料

```
root@localhost:/var/www/html/85c# php artisan queue:failed
```

ID	Connection	Queue	Class	Failed At
3	database	default	App\Jobs\ProcessPodcast	2018-04-16 13:16:05
2	database	default	App\Jobs\ProcessPodcast	2018-04-16 13:15:29
1	database	default	App\Jobs\ProcessPodcast	2018-04-16 13:15:29

在terminal下執行 `php artisan queue:retry 1` or `php artisan queue:retry all` 會將 id 為1的工作 或 所有的工作 重新進jobs那個表格內等待執行

```
root@localhost:/var/www/html/85c# php artisan queue:retry 1
The failed job [1] has been pushed back onto the queue!
root@localhost:/var/www/html/85c# php artisan queue:retry all
The failed job [3] has been pushed back onto the queue!
The failed job [2] has been pushed back onto the queue!
root@localhost:/var/www/html/85c#
```

由於要執行 `php artisan queue:work` (或者 執行 `php artisan queue:listen`)，則job queue內的工作才會被執行，這樣必須要手動執行很麻煩而且如果中間`php artisan queue:work` 這個指令因為某些原因被中斷或fail停止了，就必須手動重啟更麻煩，有根本不知道他哪時會被中斷。

所以要做以下步驟來處理這個問題，讓`php artisan queue:work` 可以一直處於執行狀態，或者說他在被中斷後會自動重啟。

1. 執行 `sudo apt-get install supervisor supervisor`
2. 在 `/etc/supervisor/conf.d` 目錄下 建立一個conf檔案 例如: `laravel-worker.conf` 以下為參考內容

```
[program:laravel-worker]
process_name=%(program_name)s_%(process_num)02d
command=php /var/www/html/85c/artisan queue:work
autostart=true
autorestart=true
user=root
numprocs=8
redirect_stderr=true
stdout_logfile=/var/www/html/85c/storage/logs/worker.log
```

3. 前兩個步驟完成supervisor 安裝，與conf檔案的設定，然後執行接下來的指令來讓supervisor 幫忙執行queue:work
4. `sudo supervisorctl reread`
5. `sudo supervisorctl update`
6. `sudo supervisorctl start laravel-worker:*`

參考網頁:

<https://laravel.com/docs/5.5/queues#running-the-queue-worker>

<https://stackoverflow.com/questions/42133628/laravel-queues-not-working>