

HABLAMOS DE



# La cola de tareas asíncrona Celery

```
159 # Main
160 #
161 if __name__ == "__main__":
162     s_area = str(sys.argv[1]) # '448602' area code
163     target_date = str(sys.argv[2]) # '20190901'
164
165     year = int(target_date[0:4])
166     month = int(target_date[4:6])
167     day = int(target_date[6:8])
168     target_date = datetime.date(year, month, day)
169     wb = Workbook()
170     ws = wb.active
171     ws.title = "Hotel list"
172     initializeSheet(ws)
173
174     for term in range(0,7):
175         in range(1, 7): #
176             call(s_area, target_date, str(1))
177             date.plan_list, 1, row
```



HABLAMOS DE



## ¿Qué es?

**Celery** es una librería de Python de código abierto que se utiliza para la **ejecución de tareas paralelas** de forma distribuida fuera del ciclo de solicitud-respuesta de HTTP.

HABLAMOS DE



# Utilidad

Utilizar Celery es útil para proporcionar una **experiencia de usuario rápida** cuando existe alguna tarea más compleja ejecutándose en un segundo plano.

HABLAMOS DE



# Características

- Software de **código abierto y gratuito.**
- **Fácil** de implementar y mantener.
- Procesa **millones de tareas.**
- Controla el **número de tareas.**
- Se integra con **frameworks** como Pyramid o Django.



# Mejores prácticas

- Asignar un **nombre** a las tareas.
- Definir siempre la **cola de trabajos**.
- Escribir tareas **cortas y concisas**.
- Utilizar **clases abstractas**.
- Dar **prioridad** a una sola cola.

HABLAMOS DE



# Conclusión

**Celery** es un administrador de trabajos en segundo plano que mejora la experiencia del usuario y ayuda a los programadores a ahorrar tiempo.



Descubre más:

[https://profile.es/blog/  
que-es-celery/](https://profile.es/blog/que-es-celery/)

