# Celery @shared_task Reference

## ■ Arguments of @shared_task

| Argument | Type | Default | Description | Example |
|---|---|---|---|---|
| name | str | module.func_name | Task name (string identifier) | @shared_task(name='tasks.add') |
| bind | bool | False | If True, passes self (Task instance) as first arg | @shared_task(bind=True) |
| max_retries | int | 3 | Max retries when calling retry() | @shared_task(max_retries=5) |
| default_retry_delay | int | 3 | Default delay before retry | @shared_task(default_retry_delay |
| autoretry_for | tuple | () | Exceptions that trigger auto-retry | @shared_task(autoretry_for=(Valu |
| retry_kwargs | dict | {} | Extra kwargs for retry() | @shared_task(retry_kwargs={'cou |
| acks_late | bool | False | Ack only after success (ensures redelivery if crash) | @shared_task(acks_late=True) |
| ignore_result | bool | False | Do not store task result | @shared_task(ignore_result=True |
| rate_limit | str | None | Rate limit execution (e.g. '10/s') | @shared_task(rate_limit='5/m') |
| time_limit | int | None | Hard execution time limit | @shared_task(time_limit=30) |
| soft_time_limit | int | None | Soft limit, raises SoftTimeLimitExceeded | @shared_task(soft_time_limit=25) |

## ■ Task Methods (when bind=True)

| Method | Arguments | Description | Example |
|---|---|---|---|
| self.retry() | exc=None, countdown=None, max_retries=None | Retries the task | self.retry(exc=e, countdow |
| self.update_state() | state=None, meta=None | Updates task state (useful for progress) | self.update_state(state=' |
| self.apply_async() | args=(), kwargs={}, countdown=None, eta=None | Runs task asynchronously | self.apply_async(countdo |
| self.apply() | args=(), kwargs={} | Runs task synchronously | self.apply(args=(2,3)) |
| self.AsyncResult() | task_id | Returns task result object | res = self.AsyncResult(se |

## ■ self.request Variables (when bind=True)

| Variable | Description | Example |
|---|---|---|
| self.request.id | Task ID | 3fa85f64-5717-4562-b3fc-2c963f66afa6 |
| self.request.args | Positional arguments | (10, 20) |
| self.request.kwargs | Keyword arguments | {'x': 10, 'y': 20} |
| self.request.retries | Number of retries so far | 2 |
| self.request.delivery_info | Broker/queue info | {'exchange': 'celery', 'routing_key': 'default'} |
| self.request.eta | Planned execution time | 2025-08-25T16:00:00 |
| self.request.hostname | Worker hostname | celery@worker1 |
| self.request.correlation_id | Tracing ID | abc123 |

■ Official Documentation: https://docs.celeryq.dev/en/stable/userguide/tasks.html