

Single Execution Task

Sometimes, you might want to have Gelato call your function only once. If so, you can make use of the `SingleExec` module which automatically cancels the task for you after one execution.

The task will still be canceled if the execution reverts on-chain. Here is how you can enable `SingleExec` when creating your task across the various task creation methods.

Gelato Automate UI

Open up the Advanced Settings panel when creating a new task and select Single execution task:

?

Gelato Automate SDK

Set `singleExec` flag to `true` when calling `createTask`.

...

```
Copy const{taskId,tx}:TaskTransaction=awaitautomate.createTask({ execAddress:counter.address, execSelector:selector,
resolverAddress:counter.address, resolverData:resolverData, dedicatedMsgSender:true, name:"Automated counter using
resolver", dedicatedMsgSender:true, singleExec:true });
```

...

Smart Contract

Include `SingleExec` module in `ModuleData.modules`. Check out the full code [here](#).

...

```
Copy ModuleData memory moduleData = ModuleData({ modules: newModule, args: newbytes });
```

```
moduleData.modules[0] = Module.PROXY; moduleData.modules[1] = Module.SINGLE_EXEC;
```

```
moduleData.args[0] = _proxyModuleArg(); moduleData.args[1] = _singleExecModuleArg();
```

```
bytes32 id = _createTask( address(this), execData, moduleData, address(0) );
```

...

[Previous](#) [Dedicated msg.sender](#) [Next](#) [Multi contract execution](#) Last updated 3 months ago On this page * [Gelato Automate UI](#)
* [Gelato Automate SDK](#) * [Smart Contract](#)