

[Foundation](#) / BlockOperation

Class

BlockOperation

An operation that manages the concurrent execution of one or more blocks.

iOS 4.0+ | iPadOS 4.0+ | Mac Catalyst 13.1+ | macOS 10.6+ | tvOS 9.0+ | visionOS 1.0+ | watchOS 2.0+

```
class BlockOperation
```

Overview

The [BlockOperation](#) class is a concrete subclass of [Operation](#) that manages the concurrent execution of one or more blocks. You can use this object to execute several blocks at once without having to create separate operation objects for each. When executing more than one block, the operation itself is considered finished only when all blocks have finished executing.

Blocks added to a block operation are dispatched with default priority to an appropriate work queue. The blocks themselves should not make any assumptions about the configuration of their execution environment.

For more information about blocks, see [Blocks Programming Topics](#).

Topics

Managing the Blocks in the Operation

```
convenience init(block: () -> Void)
```

Creates and returns an `NSBlockOperation` object and adds the specified block to it.

```
func addExecutionBlock(() -> Void)
```

Adds the specified block to the receiver's list of blocks to perform.

```
var executionBlocks: [() -> Void]
```

The blocks associated with the receiver.

Relationships

Inherits From

Operation

Conforms To

CVarArg

CustomDebugStringConvertible

CustomStringConvertible

Equatable

Hashable

NSObjectProtocol

Sendable

SendableMetatype

See Also

Operations

```
class OperationQueue
```

A queue that regulates the execution of operations.

```
class Operation
```

An abstract class that represents the code and data associated with a single task.