

# op (/js/op).lockForUpdate

```
op.lockForUpdate(  
  [lockUriCol as String]  
) as ModifyPlan
```

## Summary

Gets an early lock on documents that will be updated later in the pipeline with an operation like `remove()` or `write()`. Documents are always locked, but for complex pipelines, it can be beneficial to lock the document early, so to minimize the potential for lock contention.

Parameters	
lockUriCol	If this column is not specified then it assumes a column 'uri' is present. This can be a string of the uri column name or an op.col (./op.col). Use op.viewCol (./op.viewCol) or op.schemaCol (./op.schemaCol) if you need to identify columns in the two views that have the same column name.

## See Also

- Optic API for Multi-Model Data Access (/guide/app-dev/OpticAPI#id\_93259) in the *Application Developer's Guide*

## Example

```
declareUpdate();  
const op = require('/MarkLogic/optic');  
  
op.fromDocDescriptors([uri:"/test.xml"])  
  .lockForUpdate()  
  .execute();
```

## Example

```
declareUpdate();  
const op = require('/MarkLogic/optic');  
  
op.fromDocDescriptors([uri:"/test.xml"])  
  .lockForUpdate("uri")  
  .execute();
```

## Example

```
declareUpdate();  
const op = require('/MarkLogic/optic');  
  
op.fromDocDescriptors([uri:"/test.xml"}], "myQualifier")  
  .lockForUpdate(op.viewCol("myQualifier", "uri"))  
  .execute();
```