src - en este directorio se almacena el código fuente de las funciones lambda

con las que se va a trabajar.

\_\_init\_\_.py

create.py 🡪 lambda de creación de tarea

decimalencoder.py 🡪 workaround para el fallo conocido que no permite manejar decimales a json.

delete.py 🡪 lambda de eliminación de tarea.

get.py 🡪 lambda que muestra una tarea pasada por parámetro.

list.py 🡪 lambda que muestra todas las tareas.

requirements.txt 🡪 fichero de requerimiento de python

todoList.py 🡪 programa que genera los datos de ejemplo

update.py 🡪 lambda que actualiza una tarea

test - Tests unitarios y de integración.

[integration](https://github.com/jgarrigam/todo-list-aws/tree/1.0.0/test/integration)/todoApiTest.py 🡪 Pruebas de integración

[unit](https://github.com/jgarrigam/todo-list-aws/tree/1.0.0/test/unit)/TestToDo.py🡪 Pruebas unitarias

pipelines - pipelines de Jenkins que permiten construir el CI/CD.

[PIPELINE-FULL-CD](https://github.com/jgarrigam/todo-list-aws/tree/1.0.0/pipelines/PIPELINE-FULL-CD)

[PIPELINE-FULL-PRODUCTION](https://github.com/jgarrigam/todo-list-aws/tree/1.0.0/pipelines/PIPELINE-FULL-PRODUCTION)

[PIPELINE-FULL-STAGING](https://github.com/jgarrigam/todo-list-aws/tree/1.0.0/pipelines/PIPELINE-FULL-STAGING)

[common-steps](https://github.com/jgarrigam/todo-list-aws/tree/1.0.0/pipelines/common-steps) 🡪 contiene los ejecutables de construcción, deploy e integración de sam

template.yaml - Template que define los recursos AWS de la aplicación.

samconfig.toml - Configuración de los stacks de Staging y Producción.

localEnvironment.json - Permite el despliegue en local de la aplicación sobreescribiendo el endpoint de dynamodb para que apunte contra el docker de dynamo