



PONTIFICIA UNIVERSIDAD CATÓLICA DE CHILE
ESCUELA DE INGENIERÍA
DEPARTAMENTO DE CIENCIA DE LA COMPUTACIÓN

IIC3745: Testing

Actividad 1: Unit Testing & Coverage

Juan Pablo Sandoval & Maximiliano Narea

Martes 13-Agosto-2024

1 Objetivos

El objetivo de esta actividad es poner a prueba el conocimiento adquirido en clase sobre: cobertura de código, pruebas unitarias y mocks. Se espera que los estudiantes sean capaces de diseñar y ejecutar tests unitarios efectivos, utilizando mocks según sea necesario, para lograr una cobertura de código del 100% en los módulos proporcionados. Además, se espera que los estudiantes comprendan cómo utilizar herramientas de cobertura para identificar áreas no cubiertas y mejorarlas.

2 Descripción

En esta tarea, deberá crear pruebas unitarias para dos clases: *APIClient* y *ComplexCalculator*. Cabe destacar que el código de estas clases no cuenta con un diseño óptimo; sin embargo, el objetivo es crear pruebas unitarias sin modificar el código original.

- *ComplexCalculator*: Cree pruebas unitarias para obtener una cobertura del 100% en esta clase. Considere que la clase *ComplexCalculator* tiene diferentes flujos de ejecución (branches), por lo que deberá crear una prueba separada para evaluar cada flujo.
- *APIClient*: Cree pruebas unitarias para obtener una cobertura del 100% en esta clase. Esta clase utiliza una API externa, por lo que deberá emplear la biblioteca *mocks* de Python para evitar que sus pruebas unitarias dependan de servicios externos.

3 Código Base

GitHub Classroom. Para realizar la actividad, debe unirse al repositorio de GitHub Classroom de la sección del curso y aceptar la actividad uno.

- *Sección 1*: <https://classroom.github.com/a/SRmmCL7S>
- *Sección 2*: <https://classroom.github.com/a/ERHh-cCv>

Lo anterior le creará un repositorio con la siguiente estructura de carpetas para la actividad:

```
src/api_client.py
src/complex_calculator.py
tests/test_api_client.py
tests/test_complex_calculator.py
Actividad01.ipynb
```

Los archivos en la carpeta `tests` no contienen pruebas unitarias, por lo que deberá crearlas siguiendo la descripción de la tarea.

Cobertura y evaluación. El archivo Jupyter contiene algunos scripts para ejecutar la tarea desde Jupyter. No debe modificar las últimas celdas del Jupyter Notebook, ya que estas nos ayudan a autocorregir la tarea. También es posible ejecutar las pruebas y calcular la cobertura en consola utilizando los siguientes comandos:

```
!coverage run --rcfile=.coveragerc -m unittest discover -s tests
```

Este otro comando le mostrará qué instrucciones ya fueron cubiertas, cuáles no y el porcentaje de cobertura total alcanzado hasta el momento:

```
!coverage report -m
```

Importante: Su nota será calculada según el porcentaje de coverage total obtenido en su entrega. Este valor considera tanto porcentaje de statements como de branch coverage. Al interior de su jupyter notebook podrá encontrar una referencia para entender el cálculo de su nota.

4 Entregable, Fecha de Entrega y Penalizaciones

Deberá renombrar el archivo `Actividad01.ipynb` con su número de alumno (Ej: `17626465.ipynb`) y subir el código de los archivos de tests modificados al repositorio asignado para su tarea.

- **Fecha de Entrega:** Miércoles 14 de agosto hasta las 23:59.
- **Entrega con un día de atraso (-1 punto):** Viernes 16 de agosto hasta las 23:59. No contamos el jueves porque es feriado.
- **Entrega con dos días de atraso (-2 punto):** Sábado 17 de agosto hasta las 23:59. No contamos el jueves porque es feriado.