Trabajo Autónomo: Refactoring

# Objetivos:

* Relacionar malos olores de programación con, las técnicas de refactorización para su posterior aplicación en el proyecto.

# Referencias

* + <https://sourcemaking.com/refactoring>
  + <https://refactoring.guru/refactoring/smells>

# Generar reporte en EXCEL

* Utilice las referencias bibliográficas para llenar una tabla con la siguiente información:
  + Nombre del mal olor: inglés y español
  + Descripción corta: donde indique porque eso es un problema en el código.
  + Lista de técnicas de refactorización: con las consideraciones necesarias para aplicar cada técnica y corregir el mal olor.
  + Enlace a GitHub: Ejemplo (dentro de su propio repositorio) para demostrar el mal olor y el código corregido aplicando una de las técnicas de refactorización. No puede ser el mismo que el dado en las páginas de referencia.
* Indicar referencias adicionales utilizadas para obtener mayor información y/o ejemplos. **Utilice formato APA.**
* Lista de malos olores:
  + Long Method, Long Class, Primitive Obsession, Data clumps
  + Switch statements, Refused Bequest
  + Comments, Duplicate Code, Data Class
  + Feature envy, Inappropriate Intimacy, Shotgun Surgery

# Entregable

* Archivo Excel con los datos de los malos olores.
* 12 ejemplos de corrección de malos olores y aplicación de técnicas de refactorización en GitHub de su propia cuenta.
* Indicar las referencias utilizadas para este trabajo con formato APA. En caso de ser ejemplos propios, colocar que son de propia autoría.

# Referencias

*Cursada*. (2012). Obtenido de https://sites.google.com/site/utndesign/cursos/cursadas-anteriores/miercoles-a-la-noche-2012/seguimientoclasesmienoche2012/refactoring

Harrison, N. (27 de Abril de 2009). *RedGate*. Obtenido de https://www.red-gate.com/simple-talk/dotnet/.net-framework/exploring-smelly-code/

*Wikipedia*. (23 de Marzo de 2017). Obtenido de https://en.wikipedia.org/wiki/Data\_Clump\_(Code\_Smell)