### **1. LoadData**

Этот класс отвечает за загрузку и предобработку данных из CSV-файлов.

* В конструкторе (\_\_init\_\_) он читает тренировочный и тестовый наборы данных из файлов.
* Метод fill\_na\_lower:
  + Приводит имена столбцов к нижнему регистру.
  + Преобразует категориальный признак gender в числовой формат (1 — мужской, 0 — женский).
  + Заполняет пропущенные значения в категориальных столбцах модой (самым частым значением).
  + Заполняет пропуски в числовых столбцах медианой.

Таким образом, этот класс подготавливает данные для дальнейшего обучения и предсказания.

### **2. SplitHeart**

Наследует LoadData и отвечает за разделение данных на признаки и целевую переменную.

* При инициализации вызывает методы загрузки и предобработки из LoadData.
* Метод split:
  + Отделяет целевую переменную heart attack risk (binary) от признаков в тренировочном наборе.
  + Удаляет лишние колонки (unnamed: 0, id), которые не нужны для обучения.
  + Возвращает тренировочные признаки (X), целевую переменную (y), тестовые признаки (X\_heart\_test) и идентификаторы из тестового набора (ids).

Этот класс структурирует данные для удобного обучения модели и получения предсказаний.

### **3. GBC**

Обёртка над моделью градиентного бустинга (GradientBoostingClassifier из sklearn).

* В конструкторе создаёт экземпляр модели с заданными гиперпараметрами (learning\_rate, max\_depth, n\_estimators, random\_state).
* Метод fit обучает модель на тренировочных данных.
* Метод predict возвращает бинарные предсказания риска сердечного приступа на основе вероятностей и заданного порога (threshold).

Этот класс инкапсулирует логику работы с моделью машинного обучения.

### **Итог**

* LoadData — загрузка и очистка данных.
* SplitHeart — подготовка и разделение данных на признаки и целевую переменную.
* GBC — создание, обучение и предсказание модели градиентного бустинга.