* 0p10s: Cài đặt thư viện để trực quan hoá (matplotlib)
* 0p40s: import các thư viện cần thiết (numpy, matplotlib, math, PIL, os)
* 1p35s: khai báo hàm tính giá trị
* 2p05s: khai báo hàm tính đạo hàm
* 2p34s: vẽ đồ thị hàm f(x) trong khoảng (-4, 2)
* 3p35s: IDE Pycharm có sẵn tính năng fomat file python theo chuẩn PEP8
* 3p44s: khai báo 1 hình mới để vẽ đồ thị với kích thước rộng 10 inch, cao 7 inch
* 4p16s: dùng hàm show để hiển thị hình
* 6p08s: Gradient Descent theo công thức:
* 6p12s: khai báo điểm khởi đầu x, learning\_rate
* 6p44s: khai báo một giá trị epsilon cực nhỏ để so sánh giá trị y trước và sau khi cập nhật, nếu thay đổi rất nhỏ thì dừng quá trình Gradient Descent lại
* 8p05s: dùng vòng lặp While để thực hiện quá trình Gradient Descent
* 8p12s: tính giá trị f(x)
* 8p33s: cập nhật x
* 8p55s: so sánh giá trị f(x) trước đó với f(x) sau khi x được cập nhật, nếu khoảng cách nhỏ hơn epsilon thì dừng vòng lặp While
* 9p55s: làm sạch hình vẽ bằng phương thức clf
* 10p10s: đặt tiêu đề cho hình vẽ, bên trái sẽ là giá trị của x (lấy 3 giá trị sau dấu phảy), bên phải sẽ là số lần cập nhật i
* 11p17s: vẽ hàm f(x) trong khoảng (-4, 2)
* 11p28s: đánh dấu giá trị x hiện tại bằng phương thức scatter
* 11p44s: sau khi vẽ thì không hiển thị mà lưu lại, đối số truyền vào là tên ảnh – thay đổi theo i
* 12p05s: tính lại giá trị Gradient và tăng i lên 1 đơn vị, tiếp tục quá trình
* 12p20s: chạy thử
* 12p50s: thay đổi giá trị learning\_rate lớn hơn để quá trình Gradient Descent có thể diễn ra nhanh hơn
* 13p35p: lưu các ảnh thành 1 gif
* 17p10s: xoá ảnh sau khi có gif
* 17p41s: dùng lệnh pwd để lấy đường dẫn folder hiện tại, dán đường dẫn đó vào tham số phương thức listdir