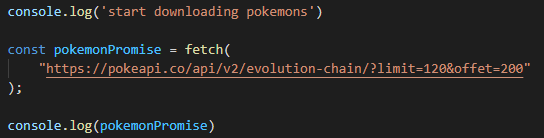
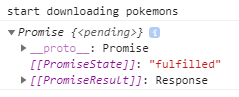
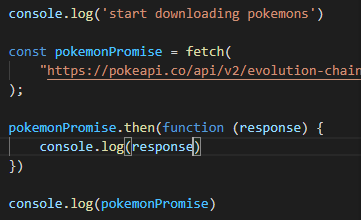
Promise:



* Kq

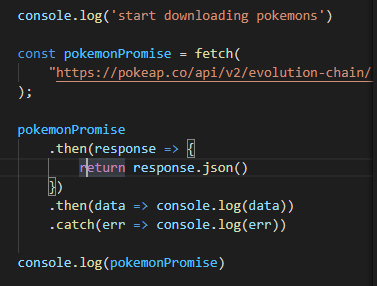


* Sử dụng Promise



.then() tức là bằng cái return của cái trước dùng nó. .then() là trả về thành công(resolve), .catch(): trả về lỗi

* Lấy dữ liệu từ trên api về:



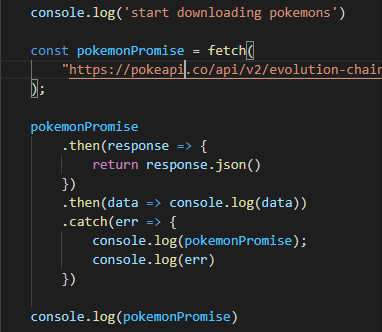
.catch() này bắt lỗi trên toàn bộ hàm chạy promise trên nó

* Chẳng hạn api bị lỗi :

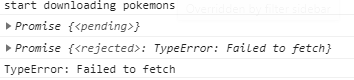


# 3 trạng thái của Promise

1. Unresolved/pending: initial state, neither fulfilled nor rejected
2. Resolved/fulfilled: complete successful



* Ví dụ bị sai api



1. rejected

# Tự tạo Promise

Ảnh có chứa văn bản

Mô tả được tạo tự động

# Hiểu rõ Promise chaining

Thực hiện nhiều asychronous lồng với nhau

- VD1:

Ảnh có chứa văn bản

Mô tả được tạo tự động

VD2:

Giống vd1 và thêm:

Ảnh có chứa văn bản

Mô tả được tạo tự động

Ảnh có chứa văn bản

Mô tả được tạo tự động

# [Challenge] Promise chaining

Link tham khảo: <https://bevacqua.github.io/promisees/>

code: tailieu\Front-ent\JS\Javascript\Document\Code mẫu\Hiểu rõ Promise Chaining.html

# Sử dụng nhiều Promise với Promise.all

Ảnh có chứa văn bản

Mô tả được tạo tự động

- Đối với Promise “avatarPromise” và “coverImagePromise” thì tùy Promise nào chạy trước thì nó sẽ show kq đó ra trước

Sử dụng Promise.all()

- Cú pháp

Ảnh có chứa văn bản

Mô tả được tạo tự động

- Run nhiều promise cùng lúc

Ảnh có chứa văn bản

Mô tả được tạo tự động

Ảnh có chứa văn bản

Mô tả được tạo tự động

Có thế chuyển thành như thế này

Ảnh có chứa văn bản

Mô tả được tạo tự động

Có thể viết chậm lại như thế này