Câu 1:

Cơ chế hoạt động JS trên trình duyệt:

* Tải mã JavaScript: Khi trình duyệt tải một trang web chứa mã JavaScript, nó sẽ tải mã và tạo ra một môi trường thực thi để chạy mã đó.
* Phân tích cú pháp (Parsing): Trình duyệt sẽ phân tích cú pháp của mã JavaScript để xác định cú pháp và cấu trúc của nó. Nếu có bất kỳ lỗi cú pháp nào, trình duyệt sẽ thông báo cho lập trình viên để sửa chữa.
* Thực thi mã JavaScript: Sau khi cú pháp được phân tích, trình duyệt sẽ thực thi từng dòng mã JavaScript một cách tuần tự. Các biểu thức và câu lệnh được đánh giá và thực hiện.
* Xử lý sự kiện và tương tác người dùng: JavaScript cho phép xử lý sự kiện trên trang web, chẳng hạn như khi người dùng nhấp chuột, gõ phím hoặc thao tác trên trang. JavaScript có khả năng tương tác với các phần tử HTML và CSS trên trang, thay đổi nội dung, kiểu dáng và thuộc tính của chúng.
* Gọi và xử lý API: JavaScript có khả năng gọi các API (Application Programming Interface) để truy cập và tương tác với các dịch vụ và tài nguyên khác nhau trên web, chẳng hạn như đọc và ghi dữ liệu từ cơ sở dữ liệu, gửi yêu cầu HTTP và xử lý dữ liệu JSON.
* Quản lý bộ nhớ: Trình duyệt quản lý bộ nhớ trong quá trình chạy mã JavaScript. Nó tự động thu gom và giải phóng bộ nhớ cho các đối tượng không còn được sử dụng, giúp tối ưu hóa hiệu suất của ứng dụng.

Câu 2: Thê nào là JS Runtime:

JavaScript runtime là môi trường mà các đoạn code JavaScript chạy trong đó và được thông dịch bởi JavaScript engine.

JavaScript Runtime Enviroment cung cấp các đối tượng, môi trường để JavaScript giao tiếp với máy tính.

Ví dụ: Node.js, Chrome Browser

Câu 3: Thế nào JS engine:

JavaScript Engine là một chương trình máy tính thực thi các đoạn code JavaScript (JS).

JavaScript Engine đọc các đoạn mã JavaScript rồi chuyển nó sang mã máy để máy tính (hoặc phần mềm máy tính như trình duyệt web, server node.js…) có thể hiểu và chạy được.

Một số JavaScript Engine phổ biến như:

* [**SpiderMonkey**](https://en.wikipedia.org/wiki/SpiderMonkey) – Phiên bản Javascript engine đầu tiên, được dùng trên trình duyệt web đầu tiên trên thế giới – Netscape Navigator, hiện tại đang được sử dụng trên Firefox, viết bằng C và C++.
* [**Chakra**](https://en.wikipedia.org/wiki/Chakra_(JavaScript_engine)) – Là một Javascript engine cũng khá lâu đời, ban đầu được sử dụng trên Internet Explorer và biên dịch JScript, nay được dùng cho Microsoft Edge, viết bằng C++.
* [**Rhino**](https://en.wikipedia.org/wiki/Rhino_(JavaScript_engine)) – Một Engine viết hoàn toàn bằng Java, cũng có lịch sử phát triển lâu đời từ Netscape Navigator, hiện tại được phát triển bởi Mozilla Foundation.
* [**Google V8**](https://en.wikipedia.org/wiki/Chrome_V8) – Được phát triển bới Google (Chromium Project).

Câu 4: Thế nào là TDZ

Khi sử dụng let/const để khai báo biến thì khi biến đó được tham chiếu trong block trước khi được khởi tạo (initialization), lúc đó kết quả trả về sẽ là ReferenceError, bởi vì biến sẽ nằm trong một khu vực gọi là Temporal Dead Zone từ lúc bắt đầu block code cho đến khi quá trình khởi tạo được diễn ra.