當我們在 JavaScript 中透過 fetch 或 XMLHttpRequest 存取資源時，需要遵守 CORS(跨來源資源共用)。瀏覽器在發送請求之前會先發送預檢請求，確認伺服器端設定正確的 Access-Control-Allow-Methods、Access-Control-Allow-Headers 及 Access-Control-Allow-Origin 等 header，才會實際發送請求。而簡單來說 CORS 是針對不同源的請求而設置的規範。透過 JavaScript 存取非同源資源時，server 必須明確告知瀏覽器允許何種請求，只有 server 允許的請求能夠被瀏覽器實際發送，否則會失敗。而在 CORS 的規範中有兩種的方法來請求資料源，分為『簡單』和『非簡單』。我們先來探討『簡單』的部分，簡單的請求需要符合兩個條件，(1) 只能是 HTTP GET, POST or HEAD 方法 (2) 自訂的 request header 只能是 Accept、Accept-Language、Content-Language 或 Content-Type，只要不符合上列的任何一項，我們就稱之為『非簡單』的請求。簡單請求的原理就是會有一個來源，首先瀏覽器發出跨來源請求時，會攜帶一個來源的 header，表示這個請求的來源。當 server 端收到這個跨來源請求時，它可以依據「請求的來源」，亦即 Origin 的值，決定是否要允許這個跨來源請求。如果 server 允許這個跨來源請求，就可以授權給這個來源的 JavaScript 存取這個資源。而授權的方法是在 response 裡加上 “Access-Control-Allow-Origin header”。當瀏覽器收到回應時，會檢查請求中的 Origin header 是否符合回應的 Access-Control-Allow-Origin header，相符的情況下瀏覽器就會讓這個請求成功，我們也可以順利地用 JavaScript 讀取到回應；反之，則瀏覽器會將這個 request 視為是不安全的而讓他失敗，即便 server 確實收到請求也成功地回應了，但基於安全性的理由 JavaScript 中沒有辦法讀到回應，這就是簡單請求的基本流程。而在非簡單請求中，瀏覽器在發送請求之前會先發送一個 preflight request（預檢請求），其作用在於先問伺服器：是否允許這樣的請求？真的允許的話，才會把請求完整地送過去。而 Preflight request 是一個 http OPTIONS 方法，會帶有兩個 request header：Access-Control-Request-Method 和 Access-Control-Request-Headers。此時在 server 端當接到 Preflight request 時會告訴瀏覽器：我允許的方法和 header 有哪些。當瀏覽器看到跨來源請求的方法和 header 都有被列在允許的方法和 header 中，就表示可以實際發送請求了，所以 server 端的回應我們稱之為 preflight response。所以當瀏覽器收到正確的 preflight response，表示 CORS 的驗證通過，就可以送出跨來源請求了。最後一步，server 還是要回應 Access-Control-Allow-Origin header。瀏覽器會再檢查一次跨來源請求的回應是否帶有正確的 Access-Control-Allow-Origin header，若這一步也沒錯的話，跨來源請求才算正式成功，這就是 CORS 的兩種請求資料的主要方法。