

في JavaScript،

الكلمات "await" و "async" تستخدم للتعامل مع الوظائف الغير متزامنة (asynchronous) بطريقة مريحة وواضحة. تسمح بكتابة الكود بشكل متزامن مع الوظائف الغير متزامنة، مما يجعل الكود يبدو وكأنه ينفذ بطريقة خطوة بخطوة.

الكلمة المفتاحية "async" تُضاف قبل تعريف دالة لتحديد أنها تحتوي على عملية غير متزامنة. على سبيل المثال:

```
``javascript
async function fetchData() {
  // قد يحتوي هنا على كود لجلب البيانات من خادم أو
  أداء عملية غير متزامنة أخرى
  return data;
}
```

في المثال أعلاه، الدالة `fetchData` مُعلنة بواسطة الكلمة المفتاحية "async"، وهذا يعني أنها ستتعامل

مع عملية غير متزامنة.

الكلمة المفتاحية "await" تُستخدم داخل الدالة المُعلنة باستخدام "async" للانتظار حتى اكتمال عملية غير متزامنة. يمكن استخدام "await" مع أي مهمة غير متزامنة، مثل استدعاء دالة تعود بقيمة من نوع Promise. على سبيل المثال:

```
``javascript
async function fetchData() {
  const response = await
  fetch('https://.com/data');
  const data = await response.json();
  return data;
}
``
```

في المثال أعلاه، الكود يستخدم "await" للانتظار حتى يتم استرداد البيانات من خادم باستخدام `fetch`، ثم يتم تحويل الاستجابة إلى كائن JSON باستخدام `response.json()`.

استخدام "await" يجعل الكود يتوقف في هذه النقطة حتى يتم اكتمال (Promise)، ومن ثم يستمر في تنفيذ الكود اللاحق. هذا يجعل عملية كتابة الكود وفهمه أكثر سهولة