**Soal Essay**

1. Jelaskan perbedaan antara "bind()", "call()", dan "apply()" dalam konteks pengaturan nilai "this" dalam fungsi.

Jawab:

* + Metode bind() mengembalikan salinan fungsi yang diikat dengan objek tertentu sebagai nilai “this”. Ia tidak langsung menjalankan fungsi tersebut, tetapi mengembalikan fungsi baru yang dapat dijalankan nanti dengan konteks yang ditentukan.
  + Metoda call() memanggil fungsi dengan objek tertentu sebagai nilai “this”. Ia langsung menjalankan fungsi tersebut dengan argument yang diteruskan secara individu.
  + Metode apply() mirip dengan metode call(), namun argument diteruskan dalam bentuk array.

1. Soal: 2. Apa itu "memoization" dalam konteks pemrograman JavaScript? Jelaskan bagaimana memoization dapat meningkatkan performa fungsi.

Jawab:

Memoization adalah teknik dalam pemrograman di mana hasil dari suatu fungsi tersimpan dalam cache (memori) setelah fungsi tersebut dipanggil dengan argument tertentu. Memoization dapat meningkatkan performa fungsi dengan mengurangi waktu eksekusi dan menghindari pengulangan perhitungan yang sama. Dengan menyimpan hasil yang sudah diketahui sebelumnya, fungsi dapat mengembalikan hasil secara langsung dari cache tanpa harus menjalankan logika atau komputasi yang kompleks. Hal ini sangat berguna Ketika fungsi tersebut sering dipanggil dengan argument yang sama.

1. Soal: 3. Jelaskan perbedaan antara "synchronous" dan "asynchronous" dalam konteks pemrograman JavaScript.

Jawab:

* + Synchronous: Dalam pemrograman synchronous, setiap pernyataan dieksekusi secara berurutan, satu per satu. Pernyataan berikutnya hanya dieksekusi setelah pernyataan sebelumnya selesai dieksekusi. Ini berarti program akan menunggu operasi yang membutuhkan waktu lama (seperti permintaan jaringan) untuk selesai sebelum melanjutkan eksekusi kode berikutnya.
  + Asynchronous: Dalam pemrograman asynchronous, kode dieksekusi tanpa harus menunggu operasi yang membutuhkan waktu lama selesai. Sebaliknya, operasi tersebut dikerjakan di latar belakang dan kode berikutnya dieksekusi. Hasil dari operasi asinkron dapat ditangani secara terpisah menggunakan callback, promise, dan async/await.

1. Soal: 4. Apa itu "generator" dalam JavaScript? Jelaskan bagaimana cara kerja generator.

Jawab:

Dalam JavaScript, generator adalah fungsi khusus yang menghasilkan serangkaian nilai secara bertahap. Mereka memungkinan kita untuk menghentikan dan melanjutkan eksekusi fungsi, serta menghasilkan nilai pada setiap titik hentian. Cara kerja generator cukup sederhana, saaa generator dipanggil, ia tidak langsung mengeksekusi kode di dalamnya seperti fungsi biasa. Sebaliknya, generator mengembalikan objek iterator yang dapat digunakan untuk mengontrol aliran eksekusi.

1. Soal: 5. Jelaskan perbedaan antara "Promise" dan "async/await" dalam JavaScript.

Jawab:

Promise adalah objek yang mempresentasikan nilai yang akan tersedia di masa depan, baik itu berhasil atau gagal, sedangkan async/await adalah fitur yang memudahkan penulisan kode asinkron dengan menggunakan sintak yang lebih mirip dengan sinkron.

**Soal Studi Kasus**

1. Buatlah sebuah form input yang memungkinkan karyawan untuk memasukkan tugas baru.
2. Tampilkan daftar tugas yang sudah ditambahkan dengan menggunakan elemen HTML yang sesuai.
3. Berikan opsi untuk menandai tugas sebagai selesai atau belum selesai.
4. Tambahkan fitur penghapusan tugas dari daftar.
5. Tampilkan jumlah total tugas yang belum selesai.
6. Tampilkan pesan atau notifikasi jika karyawan telah menyelesaikan semua tugas.
7. Buatlah tombol untuk menghapus semua tugas sekaligus.

