

## AI Response:

Angular Signals is a new state management feature introduced in Angular that p

### ### Key Concepts of Angular Signals

1. **Reactivity**: Angular signals allows you to create reactive variables that can
2. **Signals**: A signal is a primitive that represents a piece of state. You can cre

```
```typescript
import { signal } from '@angular/core';

const count = signal(0); // Creates a signal with an initial value of 0
```
```

3. **Reading and Updating Signals**:
  - You can read the current value of a signal using the signal as a function.
  - You can update the value of a signal by calling the signal with a new value or

```
```typescript
// Reading the signal
const currentCount = count(); // Gets the current value of count

// Updating the signal
count.set(1); // Updates count to 1
```
```

4. **Computed Signals**: These are derived from other signals. When a depend

```
```typescript
const doubleCount = computed(() => count() * 2);
```
```

5. **Stale Signals**: Signals can be marked as stale, which allows you to define
6. **Integration with Angular**: Angular Signals can be used alongside reactive f

### ### Benefits

- **Easier State Management**: Signals provide an easier way to manage state v
- **Less Boilerplate**: The use of signals reduces the amount of boilerplate code
- **Performance**: Signals can provide better performance as their reactive natu

### ### Example Usage