Alternative Injection Syntax

Injecting services (or, in general: dependencies) into components via the **constructor functions** is the most common way of perform such injections. You'll see this approach in **most Angular projects** you'll be working on.

However, there also is an alternative way of injecting dependencies: Via Angular's `inject()` function.

Instead of injecting `LoggingService` like this:

```
@Component(...)
export class AccountComponent {
  // @Input() & @Output() code as shown in the previous lecture

  constructor(private loggingService: LoggingService) {}
}
```

you could inject it like this, by using the `inject()` function:

```
import { Component, Input, Output, inject } from '@angular/core'; // <- Add inject import

@Component(...)
export class AccountComponent {
  // @Input() & @Output() code as shown in the previous lecture
  private loggingService?: LoggingService; // <- must be added

  constructor() {
    this.loggingService = inject(LoggingService);
  }
}
```

It's totally up to you, which approach you prefer. In this course (and, as mentioned, in most projects), we'll use the constructor approach.