Dependency Injection / Design

Display users using Angular Service

app.component.html :-

```html
<app-user></app-user>
```

user.component.css :-

```css
.users {
    margin: 0 0 2em 0;
    list-style-type: none;
    padding: 0;
    width: 15em;
}
.users li {
    cursor: pointer;
    position: relative;
    left: 0;
    background-color: #EEE;
    margin: .5em;
    padding: .3em 0;
    height: 1.6em;
    border-radius: 4px;
}
.users li:hover {
    color: #607D8B;
    background-color: #DDD;
    left: .1em;
}

.users .badge {
    display: inline-block;
    font-size: small;
    color: white;
    padding: 0.8em 0.7em 0 0.7em;
    background-color:#405061;
    line-height: 1em;
    position: relative;
    left: -1px;
    top: -4px;
    height: 1.8em;
    margin-right: .8em;
    border-radius: 4px 0 0 4px;
}
```

user.component.html :-

```html
<h2  id="title"><span>Enrolled Users</span></h2>
<ul class="users">
    <li *ngFor="let user of users" id="user-{{user.id}}">
        {{user.id}} {{user.name}}
    </li>
</ul>
```

user.component.ts :-

```typescript
import { Component, OnInit } from '@angular/core';
import { UserService } from '../user.service';
import { User } from '../user.model';

@Component({
  selector: 'app-user',
  templateUrl: './user.component.html',
  styleUrls: ['./user.component.css']
})
export class UserComponent implements OnInit {
  users: User[] = [];

  constructor(private userService: UserService) { }

  ngOnInit() {
    this.users = this.userService.getUsers();
  }
}
```

app.component.ts :-

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

@Component({
  selector: 'app-root',
  templateUrl: './app.component.html',
  styleUrls: ['./app.component.css']
})
export class AppComponent {}
```

user-list.ts :-

```typescript
import { User } from './user.model';
```

user.service.ts :-

```typescript
import { Injectable } from '@angular/core';
import { User } from './user.model';

@Injectable({
  providedIn: 'root'
})
export class UserService {
  private users: User[] = [
    new User(1011, 'Abimbola'),
    new User(1012, 'Lethabo'),
    new User(1013, 'Adaeh'),
    new User(1014, 'Akello'),
    new User(1015, 'Awino'),
    new User(1016, 'Esiankiki'),
    new User(1017, 'Etemaad'),
    new User(1018, 'Jimiyu'),
    new User(1019, 'Kayode'),
    new User(1020, 'Kubwa'),
    new User(1021, 'Lebron'),
    new User(1022, 'Mansa'),
    new User(1023, 'Nomalanga'),
    new User(1024, 'Nya'),
    new User(1025, 'Rashidi'),
    new User(1026, 'Warui')
  ];

  getUsers(): User[] {
    return this.users;
  }
}
```

user.model.ts :-

```typescript
export class User {
  id: number;
  name: string;

  constructor(id: number, name: string) {
    this.id = id;
    this.name = name;
  }
}
```