Angular: App Startup (V19 and later)

During application startup, you may have activities that need to occur before the first page opens. One example is a splash screen.

> NOTE: This page applies to Angular v19 and later, as APP_INITIALIZER was deprecated in V19.
>
> See this page for how to do the same, before v19: https://wiki.galaxydump.com/link/130

To make one occur, do these things:

1. Open the app.config.ts of your app.
2. Add an import for provideAppInitializer:

```
import { provideAppInitializer } from '@angular/core';
```

3. Create an initializeApp method in the file (or, somewhere else that you can import).
   It must return a Promise<boolean>. So, make sure its signature looks like this:

```
export function initializeApp(): Promise<boolean>
```

> NOTE: The old way of doing this, used an initializer method that returned a () => Promise<boolean>
> So, be sure to update the return when migrating older code.

4. Add a provider entry for the initializer:
   It looks like this:

```
export const appConfig: ApplicationConfig = {
  providers:
  [
    provideAppInitializer(initializeApp)
  ]
};
```

5. If your initializeApp function needs to run some async logic, structure it like this:

```
export function initializeApp(): Promise<boolean>
{
  const _appsvcconfig = inject(AppconfigService);

  console.log("AppModule-" + "_" + ":initializeApp - triggered.");

  // To ensure that dependencies are stood up and ready, we must wrap our logic in a lambda, instead
of naked execution...
  return (async (): Promise<boolean> =>
  {
    console.log("AppModule-" + "_" + ":initializeApp - inside startup promise...");

    ... DO SOME THINGS...

    // Report success or failure...
    return true;
  })(); // Notice the () at the end - this immediately invokes the async lambda
}
```

Revision #2
Created 16 March 2025 02:04:05 by glwhite
Updated 16 March 2025 02:14:34 by glwhite