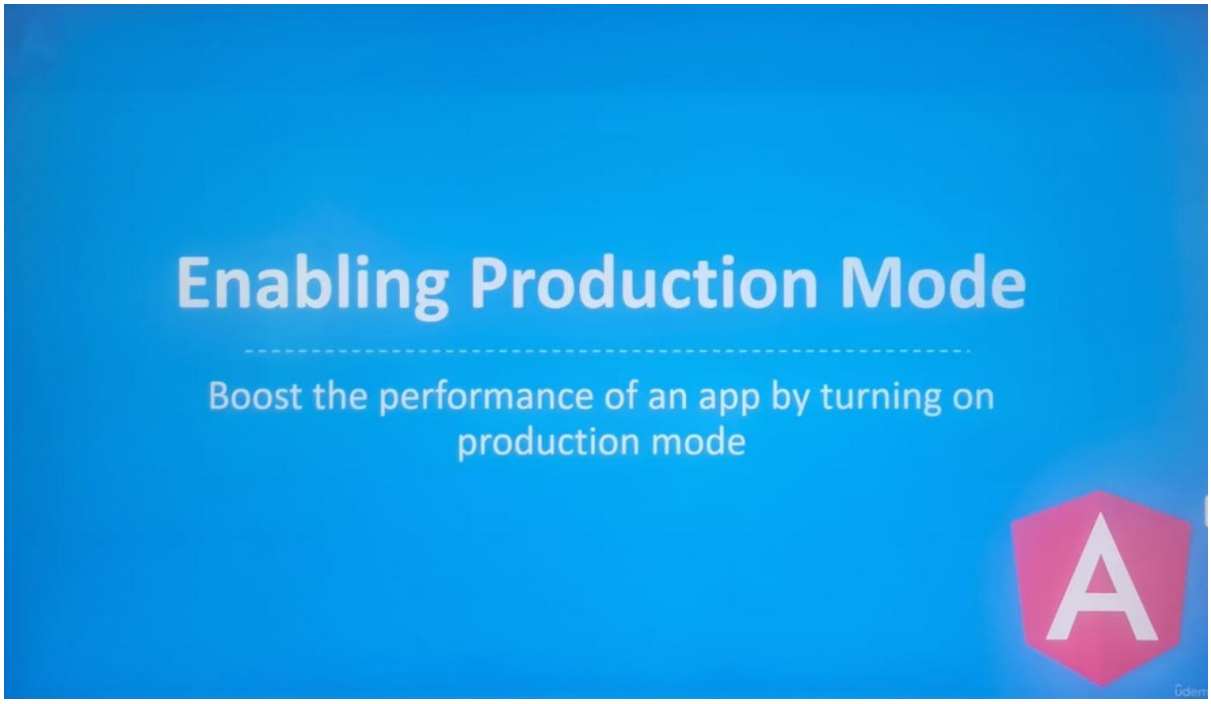
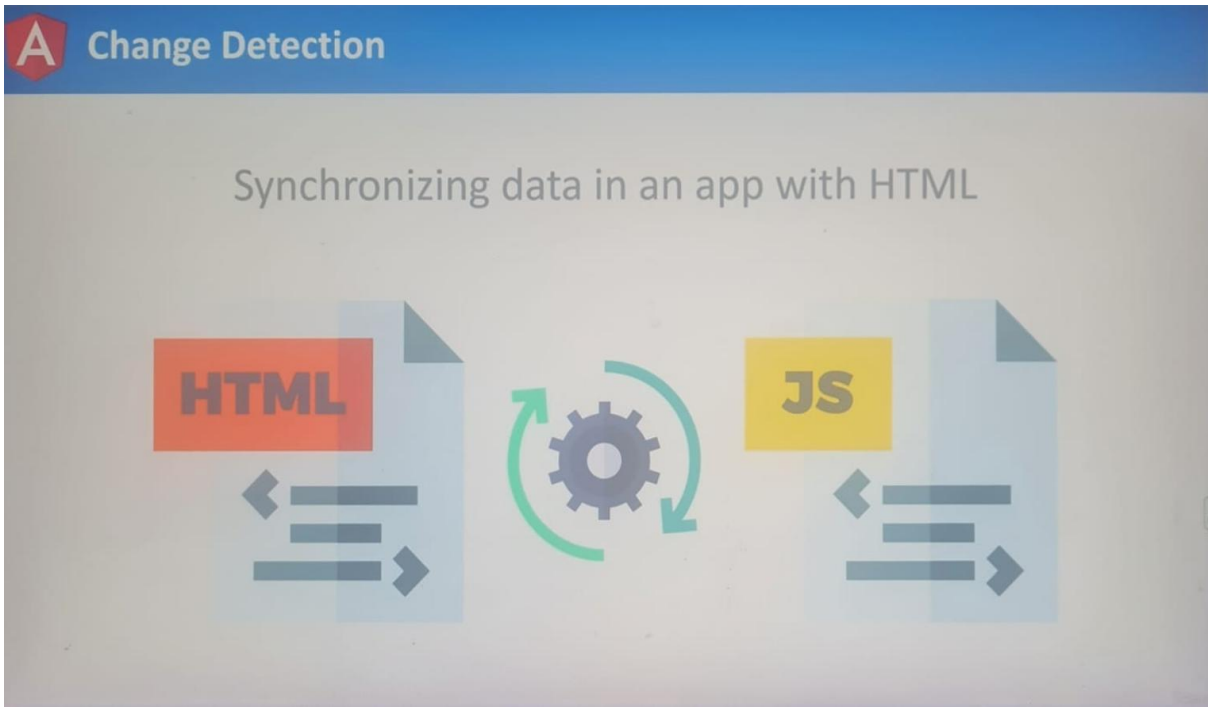
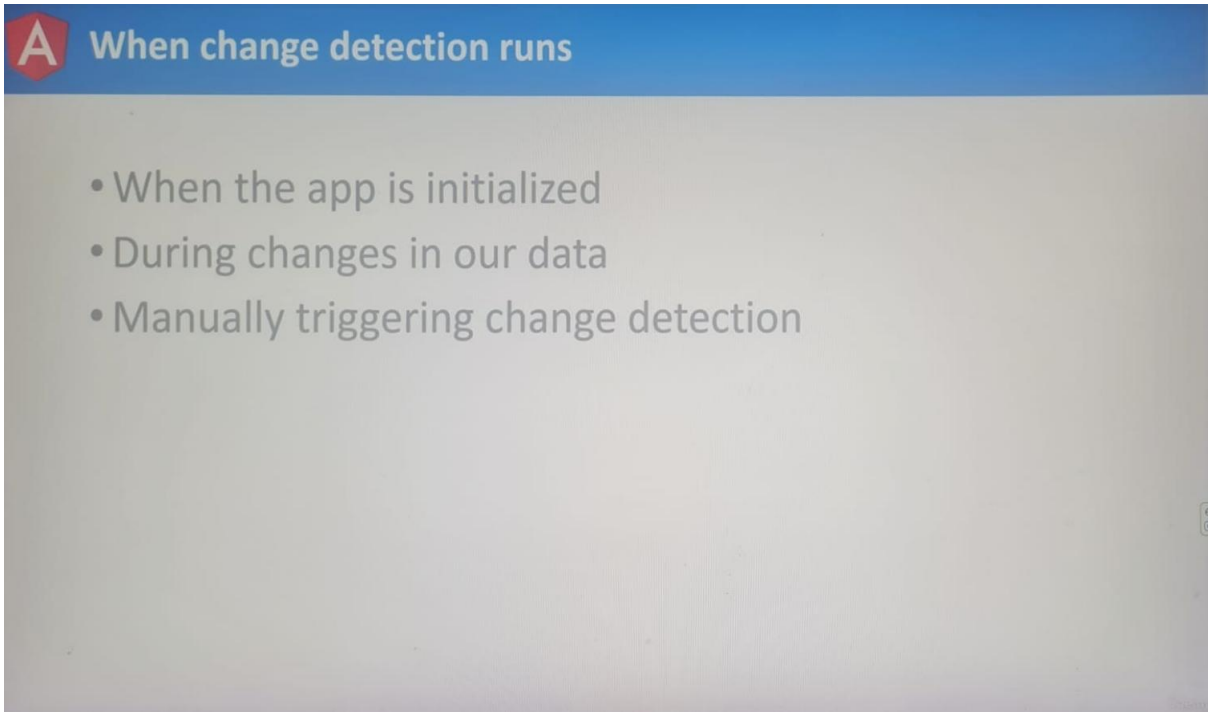
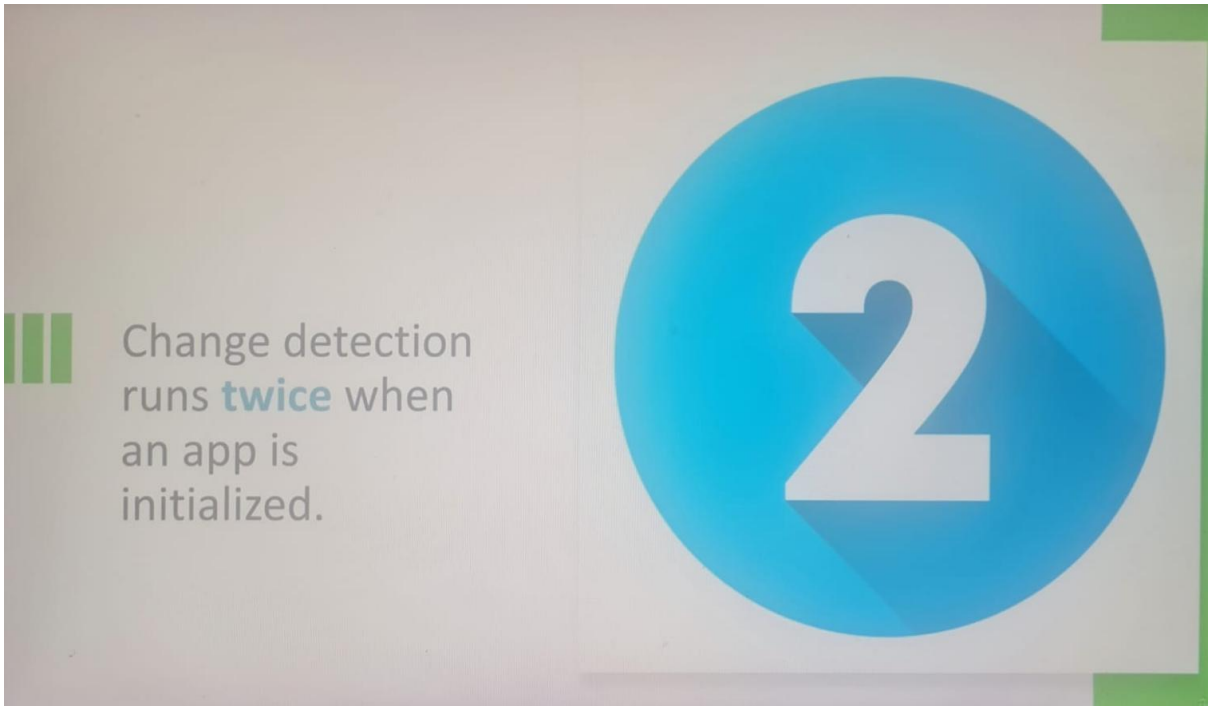
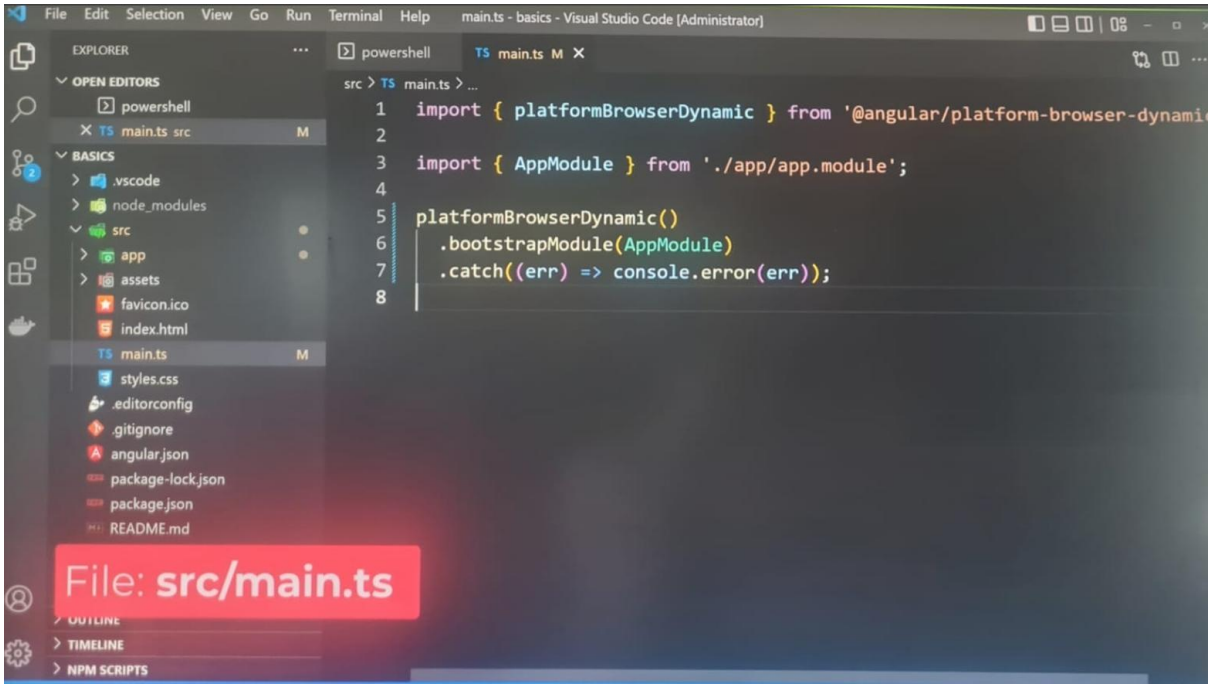
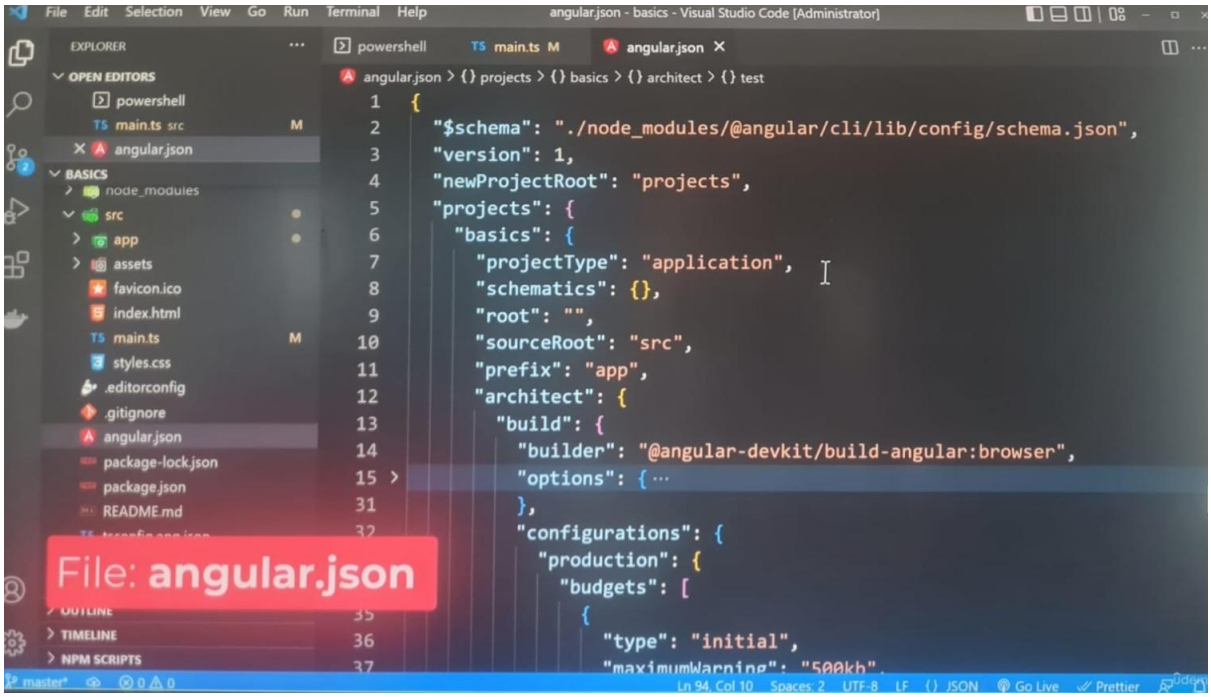

Enabling Production Mode
Boost the performance of an app by turning on production mode

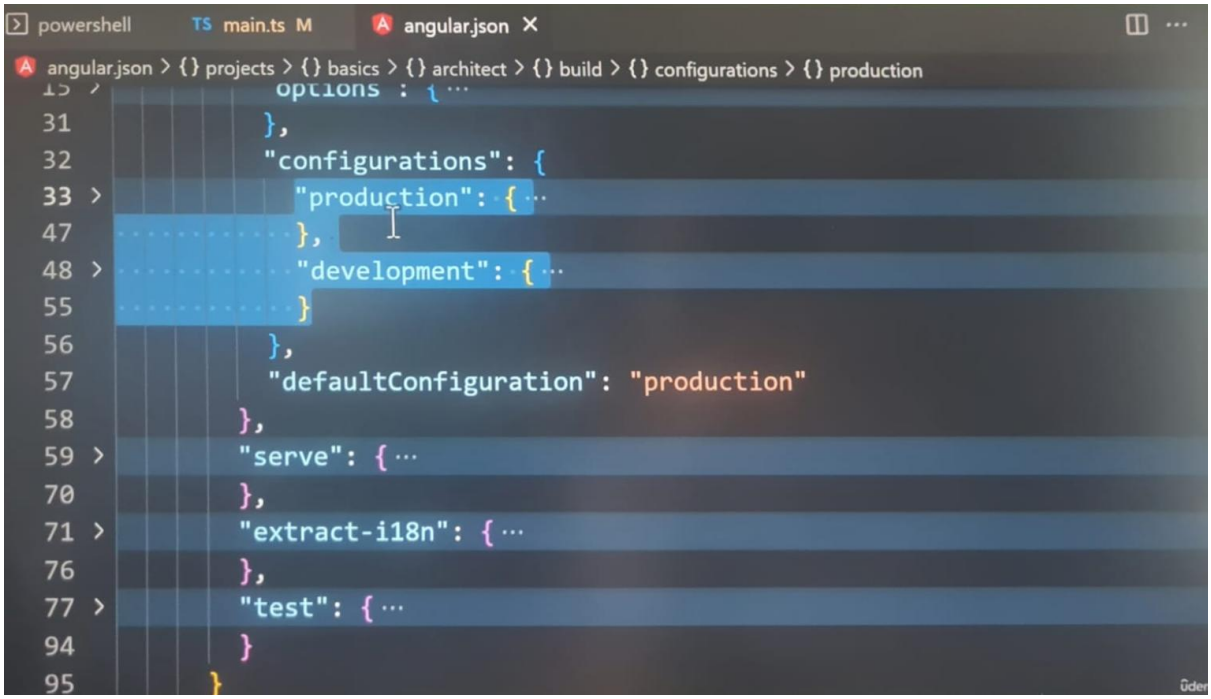

Change Detection
Synchronizing data in an app with HTML
HTML
JS

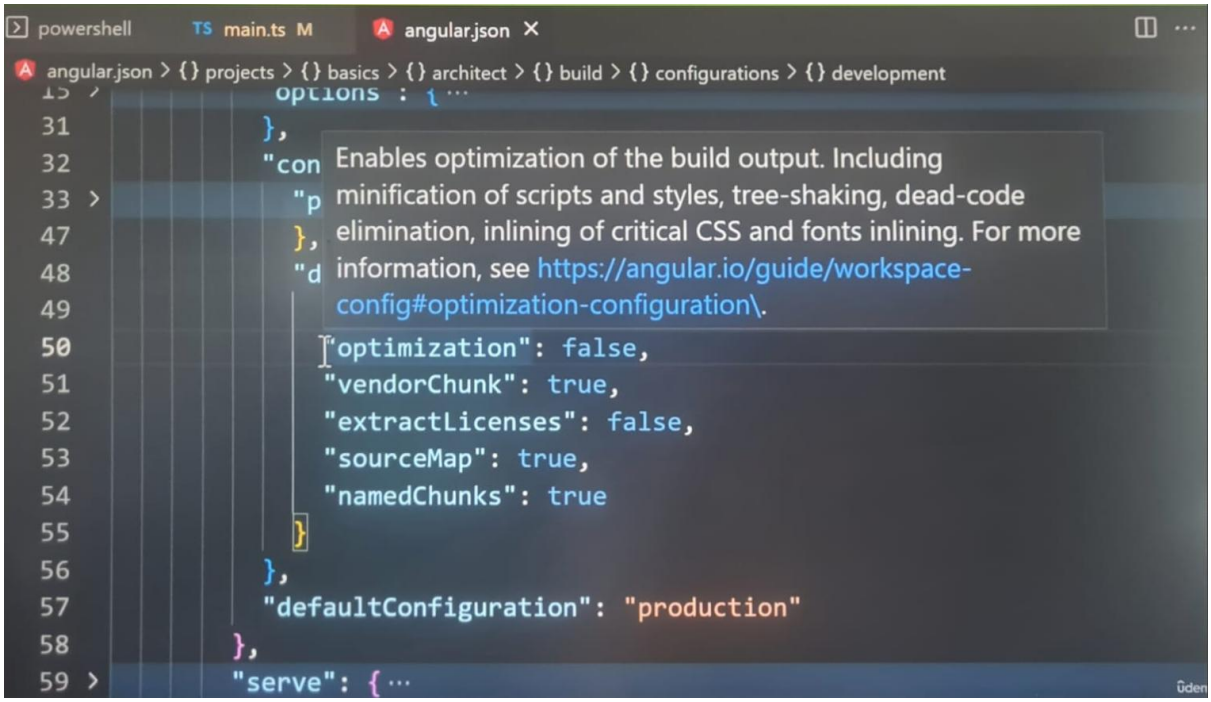

When change detection runs
• When the app is initialized
• During changes in our data
• Manually triggering change detection


Change detection runs twice when an app is initialized.
2


File: src/main.ts
import { platformBrowserDynamic } from '@angular/platform-browser-dynamic';
import { AppModule } from './app/app.module';
platformBrowserDynamic()
.bootstrapModule(AppModule)
.catch((err) => console.error(err));


File: angular.json
{
"$schema": "./node_modules/@angular/cli/lib/config/schema.json",
"version": 1,
"newProjectRoot": "projects",
"projects": {
"basics": {
"projectType": "application",
"schematics": {},
"root": "",
"sourceRoot": "src",
"prefix": "app",
"architect": {
"build": {
"builder": "@angular-devkit/build-angular:browser",
"options": {
},
"configurations": {
"production": {
"budgets": [
{
"type": "initial",


powershell
main.ts M
angular.json
angular.json > {} projects > {} basics > {} architect > {} build > {} configurations > {} production
},
"configurations": {
"production": {
},
"development": {
}
},
"defaultConfiguration": "production"
},
"serve": {
},
"extract-i18n": {
},
"test": {
}
}


powershell
main.ts M
angular.json
angular.json > {} projects > {} basics > {} architect > {} build > {} configurations > {} development
Enables optimization of the build output. Including minification of scripts and styles, tree-shaking, dead-code elimination, inlining of critical CSS and fonts inlining. For more information, see https://angular.io/guide/workspace-config#optimization-configuration\.
"optimization": false,
"vendorChunk": true,
"extractLicenses": false,
"sourceMap": true,
"namedChunks": true
},
"defaultConfiguration": "production"
},
"serve": {