

1. Napraviti index.html. Dodati funkcionalnost Angulara na stranicu. Pomoću Angulara na stranici ispisati vrednost izraza 555/33. Koraci:
 - a. Uključiti biblioteku angular.js u index.html (paziti da se angular.js biblioteka nalazi na navedenom mestu)
 - b. Dodati ng-app direktivu
 - c. Izraz 555/33 napisati kao interpolaciju {{555/33}}
2. Napraviti modul „winesModule“. Povezati napravljeni modul sa aplikacijom. Koraci:
 - a. Napraviti app.js u kome se nalazi definicija modula
 - b. Uključiti skript app.js u index.html stranicu posle angular biblioteke
 - c. Postaviti ime modula kao vrednost ng-app direktive
3. Napraviti kontroler WinesController. U kontroleru napraviti string message sa vrednošću „Aplikacija za pregled vina“. Prikazati message string kao heading na stranici index.html. Koraci:
 - a. Napraviti WinesController.js u kome se nalazi definicija kontrolera. U kontroler injektovati \$scope. Zakačiti message property za \$scope.
 - b. Uključiti skript WinesController.js u index.html posle angular biblioteke i posle app.js
 - c. Namestiti da body tag u indeksu bude kontrolisan od strane WinesController-a (iskoristiti ng-controller direktivu). U okviru body-ja dodati h1 tag u okviru koga ispisati vrednost message property-ja pomoću interpolacije.
4. Na index.html dodati jedan input (text). Pored inputa staviti dugme (button). Namestiti da kada korisnik klikne na dugme bude prikazan alert u komeje napisano šta je korisnik uneo u tekstualni input. Za alert iskoristiti Angular-ov \$window servis
 - a. Dodati input i dugme na index.html
 - b. U okviru input-a iskoristiti ng-model direktivu i postaviti njenu vrednost na „userInput“. Efekat ovoga će biti da šta god korisnik unese u input – to će biti zakačeno za \$scope. userInput u okviru kontrolera koji kontroliše dati deo stranice (u ovom slučaju to je WinesController).
 - c. U okviru dugmeta iskoristiti ng-click direktivu i postaviti njenu vrednost na „doClick()“. Kada korisnik klikne na dugme izvršiće se funkcija \$scope.doClick() koju ćemo definisati u kontroleru
 - d. U WinesController kontroleru definisati funkciju \$scope.doClick(). Kako bi se u okviru ove funkcije koristio \$window servis – injektovati ovaj servis u kontroler. U funkciji ga iskoristiti za kreiranje alert-a
5. U WinesController dodati niz vina. Prikazati ovaj niz vina u tabeli na stranici index.html
 - a. Dodati niz vina u kontroler WinesController. Zakačiti ga za \$scope (postaviti vrednost \$scope.wines)
 - b. Napraviti tabelu u index.html. Iskoristiti ng-repeat direktivu da se u tabeli prikaže niz vina
6. Sortirati tabelu sa vinima opadajuće prema *name* property-ju vina
 - a. U index.html, u ng-repeat direktivi iskoristiti orderBy filter
7. Napraviti custom servis WinesDataService. Prebaciti podatke o vinima iz kontrolera u novokreirani servis. Iskoristiti servis u kontroleru kako bi se vina i dalje prikazivala. Koraci:

- a. Kreirati WinesDataService.js : Definisati funkciju WinesDataService i registrovati je kao servis u modulu.
 - b. U okviru funkcije WinesDataService:
 - i. definisati prazan niz vina kao promenljivu
 - ii. napraviti funkciju init() u kojoj se nalaze podaci sa vinima i ti podaci se koriste da se popuni gore definisani niz vina
 - iii. napraviti funkciju getAll() koja vraća niz vina (definisan u tački i.)
 - iv. postaviti da funkcija getAll() bude “javna” – revealing module pattern
 - c. Uključiti skriptu WinesDataService.js u index.html (bilo gde posle angular.js i app.js)
 - d. Injektovati WinesDataService u WinesController
 - e. U WinesController-u pozvati funkciju getAll iz WinesDataService servisa kako bi se dobio niz vina
8. Izmeniti WinesDataService tako da se podaci dobivaju sa servera
- a. Pokrenuti node-server-Angular. U okviru foldera src/db treba da se nalazi wines.json. Da bi ste proverili da li je sve u redu, u browser-u otidite na link localhost:3000/api/wines. Treba da se izlistaju sva vina iz kolekcije.
 - b. WinesDataService
 - i. Ukloniti niz vina i init() funkciju
 - ii. U WinesDataService injektovati \$http servis
 - iii. U funkciji getAll pomoću \$http servisa napraviti GET poziv na link localhost:3000/api/wines. Rezultat funkcije neka bude povratna vrednost \$http.get() metode – promise objekat.
 - c. WinesController:
 - i. Povratna vrednost getAll() je promise – nad njom pozvati then() funkciju i registrovati onSuccessCallback i onErrorCallback.
 - ii. Funkcija onSuccessCallback prima odgovor sa servera. U property-ju *data* objekta koji se u nju prosleđuje se nalaze podaci koje je vratio server (response.data). Setite se da server vraća objekat sa 2 property-ja – results i count. U results se nalazi niz vina, a u count ukupan broj vina koji odgovara kriterijumu pretrage. Dakle, \$scope.wines = response.data.results.
 - iii. U onErrorCallback prikazati alert da podaci nisu uspešno dobavljeni sa servera. Probati i ovu varijantu, npr. ugasiti backend server i pokrenuti program. Dodati parameter reason u funkciju. U debugger-u pogledati šta server prosleđuje u ovaj parameter.