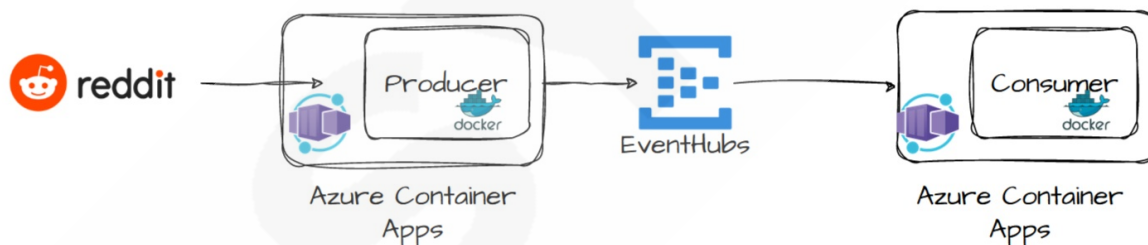


1. laboratorijska vježba

- Uvod
- Microsoft Azure
- Event Hubs
- Kreiranje Event Huba
- 1. Zadatak
- Container Apps
- 2. Zadatak

Uvod

Tijekom laboratorijskih vježbi stvorit ćete svoju platformu na Microsoft Azure oblaku, zaduženu za unos, pohranu, i transformaciju podataka. Cilj prve laboratorijske vježbe je dobiti podatke koji će se obrađivati na vašoj platformi. Na slici 1 prikazan je opseg vaše platforme tijekom prve laboratorijske vježbe. Upoznati ćete se s Microsoft Azure oblakom te servisom **Event Hubs**, uslugom za strujanje podataka. Napisat ćete **Producer** program koji podatke s Redditovog API-ja dobavlja i šalje na Event Hub te **Consumer** program koji ih čita s Event Hubs. Vaše programe stavit ćete u **Docker kontejnere**. Pomoću usluge **Container Apps**, svoje Docker kontejnere deployat ćete na Azure oblak.



Slika 1. Arhitektura projekta u 1. laboratorijskoj vježbi

Microsoft Azure

Microsoft Azure je platforma za računarstvo u oblaku koju je stvorio Microsoft. On sam ga koristi za izgradnju, razvoj i upravljanje aplikacijama kroz njihovu globalnu mrežu podatkovnih centara. Kako biste mogli nastaviti ovu vježbu, prvo što trebate napraviti je kreirati vlastiti Microsoft Azure račun. Besplatno to možete učiniti pomoću vlastitog AAI Edu računa (ako vas traži informacije o kartici, onda niste išli preko AAIedu računa). Da počnete s izradom računa, pritisnite sljedeći gumb na intranetu FER-a:

Kalendar

< **UTORAK** >
18. 4. 2023.



Nema događaja za
18. 4. 2023.

Softver

PlagScan






Matlab 

Pretplata

  Svi portleti

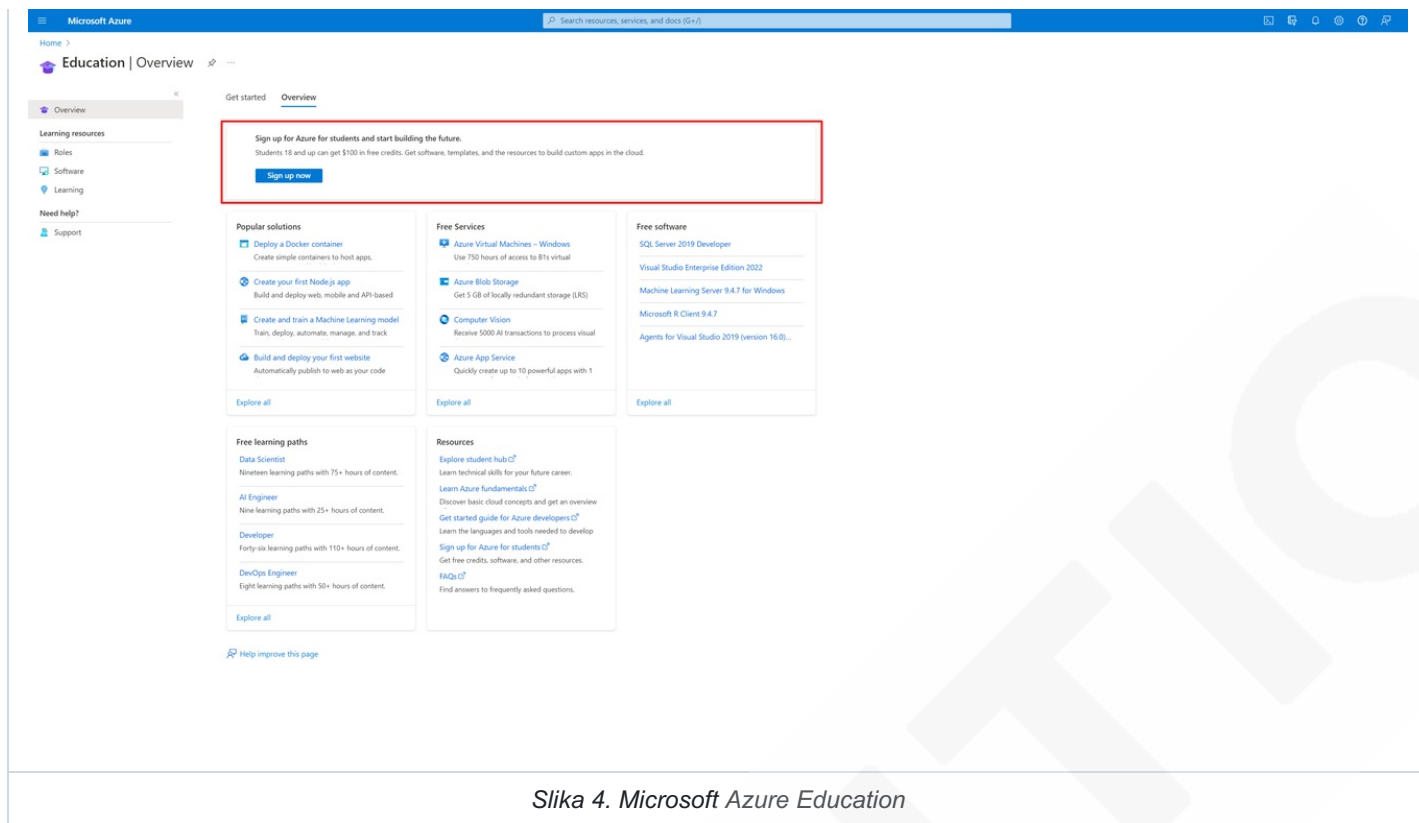
  Portlet Edu Activity scores

  Portlet Sadržaj

 Uređivanje pretplata

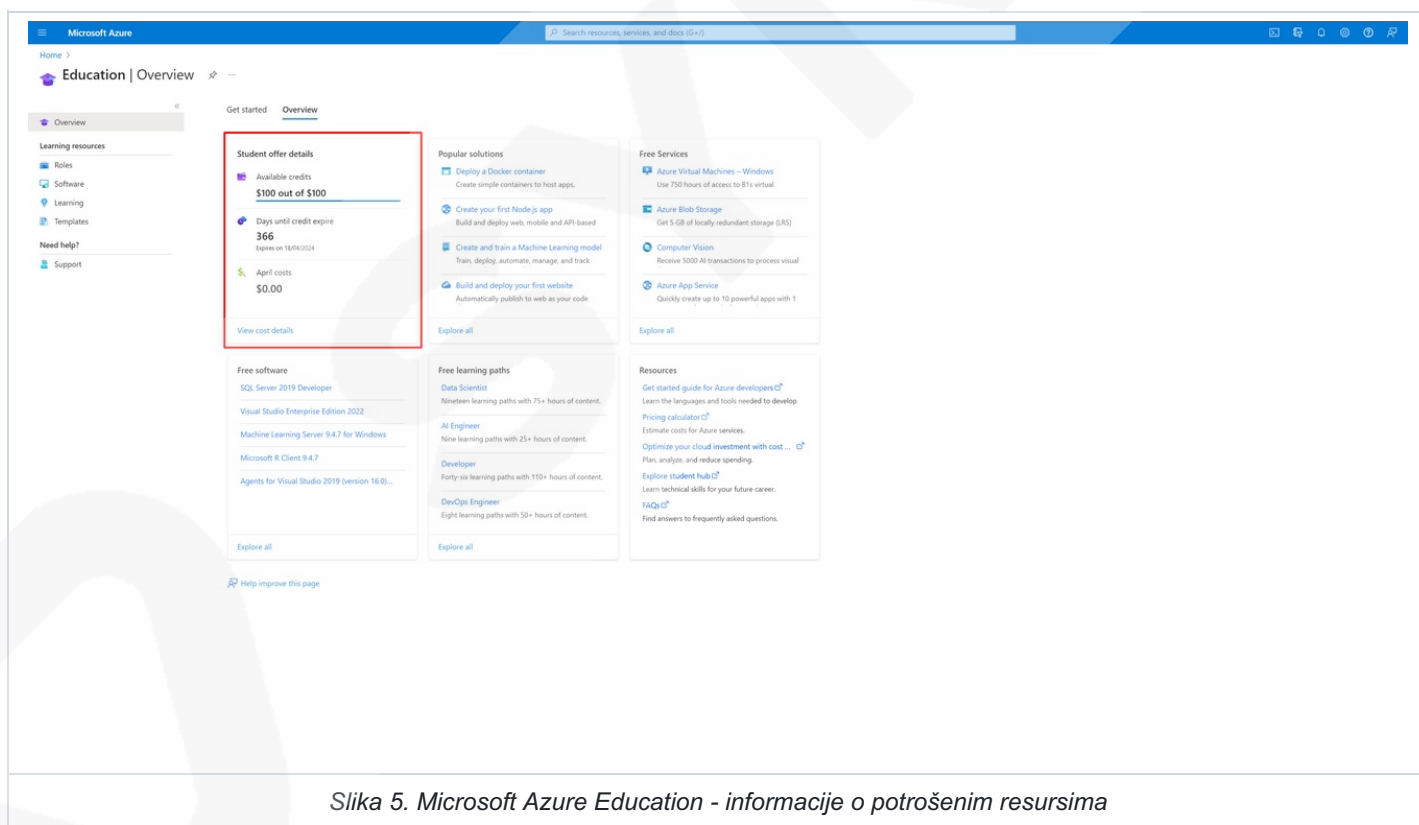
Slika 3. Microsoft Azure link na FerWebu

Na poveznici ćete se morati prijaviti da potvrdite identitet te kada se prijavite doći ćete na sljedeću stranicu:



Slika 4. Microsoft Azure Education

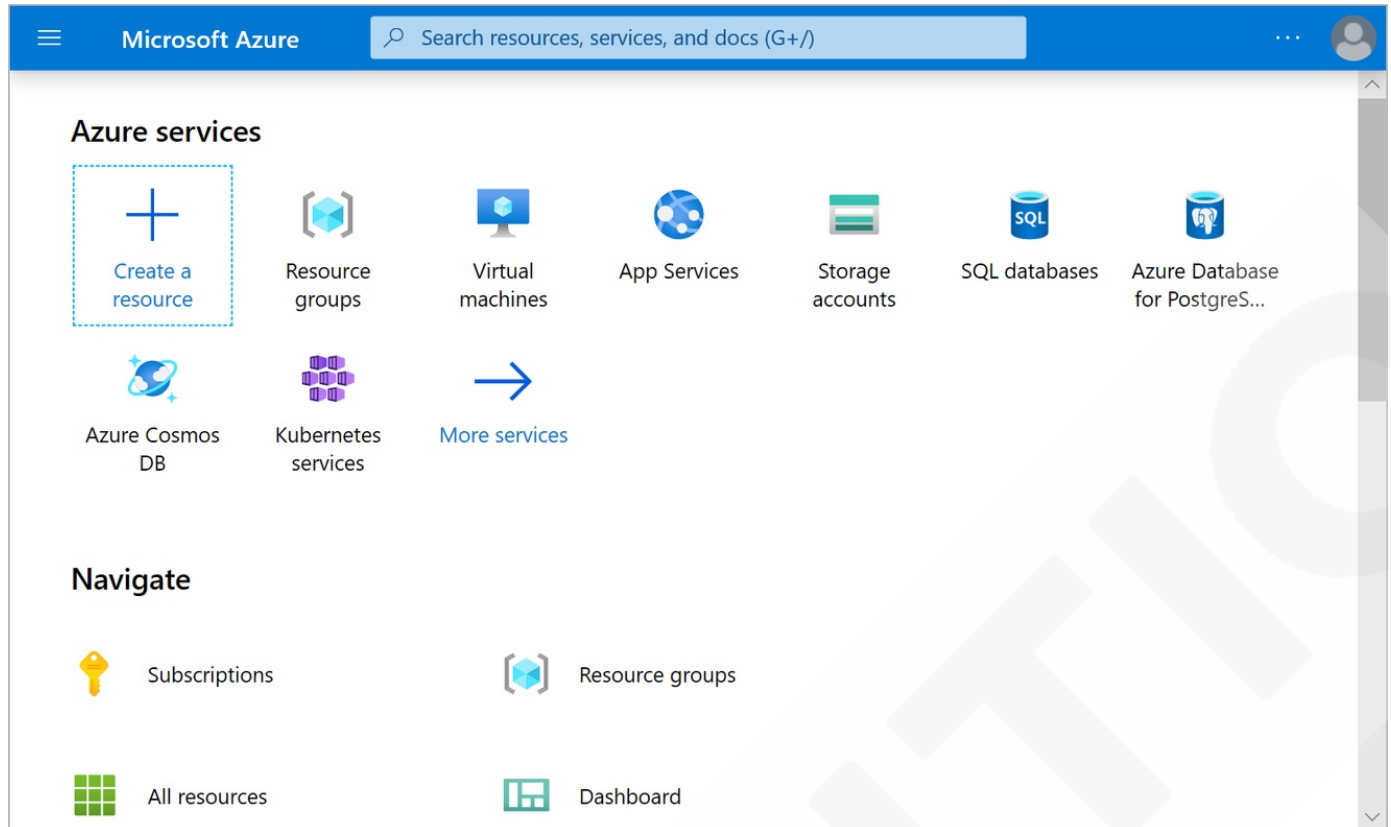
U crvenom okviru pritisnite gumb “Sign up now” te na idućoj stranici odaberite “Start for free” koji će Vas odvesti do forme koju trebate ispuniti s osobnim podacima (ime, prezime, mail, adresa...). Kada ispuniti formu, dobit ćete potvrdni mail, a gore navedena stranica sada izgleda ovako:



Slika 5. Microsoft Azure Education - informacije o potrošenim resursima

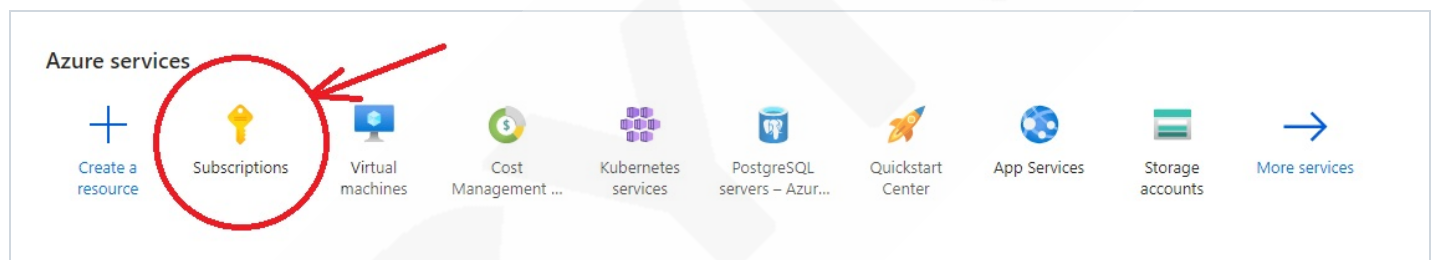
Uvod u Microsoft Azure usluge u oblaku

Sljedeća slika prikazuje Azure portal do kojeg se dolazi odabirom tri crte (gore lijevo) i zatim Home. Pomoću njega možete upravljati vašom Azure pretplatom. Za početak preporučujemo da pomoću trake za pretraživanje pokušate pronaći neke od resursa spomenutih na predavanjima kako biste uočili što sve nudi Azure oblak.



Slika 6. Microsoft Azure Home

Sada provjerite koju pretplatu (engl. subscription) koristite. Trebala bi vam biti ponuđena Azure for Students pretplata.



Slika 7. Azure Services

Kako biste mogli kreirati vlastite resurse u nastavku vježbe, potrebno je kreirati vlastitu resursnu grupu odabirom Resource groups opcije. Kliknite na gumb create, zatim odaberite prethodno spomenutu pretplatu, kao ime resursne grupe unesite **fervjestina** te za regiju odaberite West Europe.

Slika 8. Kreiranje resursne grupe

Napomena Sve resurse koje kreirate na Azure Cloudu potrebno je imenovati prema sljedećoj konvenciji: - (npr. **rg-ferlab**). Za koristite kratice (**resource group = rg, azure kubernetes service = aks** itd.). U slučaju da nazivi pojedinih resursa dopuštaju samo mala slova i brojeve, izostavite "-".

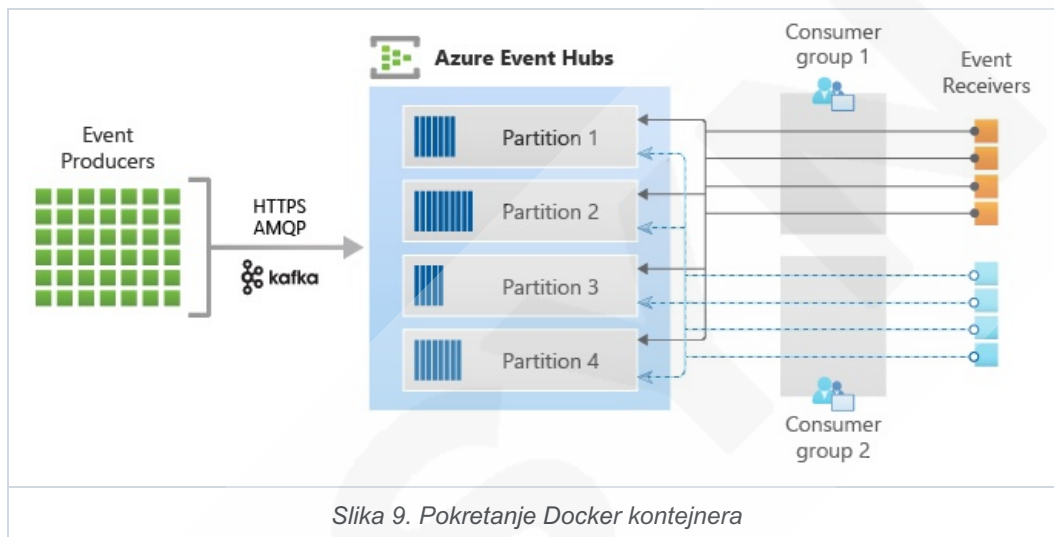
Napomena Pažljivo pratite upute jer se neki resursi na Azure cloudu **naplaćuju**. Vi kao studenti imate pravo potrošiti određenu svotu besplatno, međutim, nakon toga se naplaćuje. Na kraju uputa je napomena da resurse koji neće biti potrebni obrišete te tako i postupite.

Event Hubs

Podaci su vrijedni samo kada postoji **jednostavan način za obradu i dobivanje pravovremenih uvida iz izvora** podataka. Event Hubs pruža **distribuiranu platformu za obradu toka podataka s niskom latencijom i besprijekornom integracijom, s podacima i analitičkim uslugama unutar i izvan Azurea** za izgradnju vašeg potpunog cjevovoda velike količine podataka.

Event Hubs **predstavlja "ulazna vrata" za cjevovod događaja**, koji se u arhitekturama rješenja često naziva **ingestor događaja**. Ingestor događaja je **komponenta ili usluga** koja se nalazi **između producera događaja i consumera** događaja.

Event Hubs je **moderna platforma za strujanje velikih podataka i usluga** za unos događaja koja se može neprimjetno integrirati s drugim Azure i Microsoftovim uslugama, kao što su **Stream Analytics, Power BI i Event Grid**, s vanjskim uslugama kao što je **Apache Spark**. Usluga može obraditi milijune događaja u sekundi s malom latencijom. Podaci koji se šalju u čvorište događaja (instanca čvorišta događaja) mogu se transformirati i pohraniti korištenjem bilo kojeg pružatelja analitike u stvarnom vremenu ili adaptera za grupiranje ili pohranu.



Slika 9. Pokretanje Docker kontejnera

Event Hubs podupire Apache Kafka kao komunikacijski protokol komunikacije. U tablici ispod možete vidjeti kako se razlikuju stavke u Kafka i Event Hubs konceptima.

Kafka	Event Hubs
Cluster	Namespace
Topic	Event Hub
Partition	Partition
Consumer Group	Consumer Group
Offset	Offset

Kreiranje Event Huba

Za stvaranje Event Huba potrebno je **prvo stvoriti Event Hubs namespace**.

1. Kreiranje Event Hubs namespacea:

- u Azure portalu pronađite **Event Hubs** te kliknite **Create**
- za **Subscription** odaberite svoju studentsku pretplatu
- za **Resource group** odaberite resursnu grupu kreiranu za ovu laboratorijsku vježbu

- upišite proizvoljni naziv namespacea (**Namespace name**)
- postavite Europe West kao regiju (**Location**)
- za **Pricing tier** odaberite **Basic**
- postavite **Throughput Units** na 1
- kliknite **Review + Create** te potom **Create**

2. Kreiranje Event Huba:

- pozicionirajte se na stranici kreiranog namespacea te u lijevom izborniku u odjeljku **Entities** odaberite opciju **Event Hubs**
- pritisnite ikonu **+ Event Hub** kako biste kreirali instancu
- unesite ime Event Huba (**Name**)
- kliknite **Review + Create** te onda **Create**

API - aplikacijsko programsko sučelje

Aplikacijska programska sučelja (APIs) standardni su način razmjene podataka između sustava. Klijentski programi koriste aplikacijska programska sučelja (APIs) za komunikaciju s web uslugama. Općenito govoreći, API izlaže skup podataka i funkcija kako bi olakšao interakciju između računalnih programa i omogućio im razmjenu informacija. Web API je sučelje web usluge koje izravno sluša i odgovara na zahtjeve klijenata.

Reddit i Reddit API

Reddit je internetska platforma na kojoj korisnici objavljuju, komentiraju i pregledavaju sadržaje na različitim tematskim *subredditima*. Svaki subreddit bavi se određenom temom, a korisnici mogu objavljivati tekstualne i vizualne sadržaje te sudjelovati u raspravama.

Putem Reddit API-ja, moguće je pristupiti različitim funkcionalnostima platforme, kao što su čitanje i pisanje postova, komentiranje, upravljanje korisničkim računima te dobivanje informacija o sadržaju i korisnicima. Pruža programski način za automatizaciju interakcija s Redditom, što je korisno za izradu aplikacija, alata za analizu sadržaja, praćenje trendova i mnoge druge svrhe.

Tijekom vježbine, Reddit API služiti će kao izvor podataka koje ćete obrađivati kroz Azure platformu.

1. Zadatak

Tema ove laboratorijske vježbe je praćenje Reddit postova. Ako vam je ovo prvi susret s API-jima i Event Hubom preporučujemo da odaberete Python. Dobro proučite dokumentaciju te pronađite zahtjev(e) koji će vam pomoći u ostvarenju tog zadatka. Prije nego što krenete programirati možete se poslužiti Postmanom kako biste ustvrdili kakav zahtjev trebate poslati te kojim podacima želite pristupiti.

Idući korak vašeg zadatka je stvaranje proizvođača (producer) i potrošača (consumer). Za povezivanje na Azure Event Hubs koristite connection string. Na stranici kreiranog namespacea u lijevom izborniku odaberite opciju **Shared access policies**. Izaberite **RootManageSharedAccessKey** te kopirajte **Connection String-primary key** s kojim ćete se kasnije spajati na svoj EventHubs.

Stvaranje producera

- Odaberite i uvezite potrebne biblioteke
- Spojite se na prethodno stvoreni Event Hub
- Stvorite API fetcher koji će podatke o objavama subreddita **r/dataengineering** slati na Event Hub. Dohvatite prvih 10 objava iz kategorije **"Top"** (of All Time)
 - Za dohvaćanje podataka koristite običan HTTP request
 - Podatke s API-ja ne filtrirajte, pošaljite sve dostupne podatke (sva polja iz API odgovora) o svakoj objavi
 - Nakon što se poslali svih 10 objava na Event Hubs na kraju programa napišite beskonačnu petlju

Napomena Ovo inače nije praksa, ali s obzirom na to da se Azure resursi naplaćuju ovime ćete smanjiti mogućnost velikih troškova.

Stvaranje consumera

- Spojite se na prethodno stvoreni Event Hub

- Kada poruke pristignu, pročitajte ih u JSON formatu te ispišite, u stvarnom vremenu
- Proučite polja i podatke dostupne o objavi

Koristeći upute iz prethodnog labosa pohranite producera i consumera na GitHub repozitoriju.

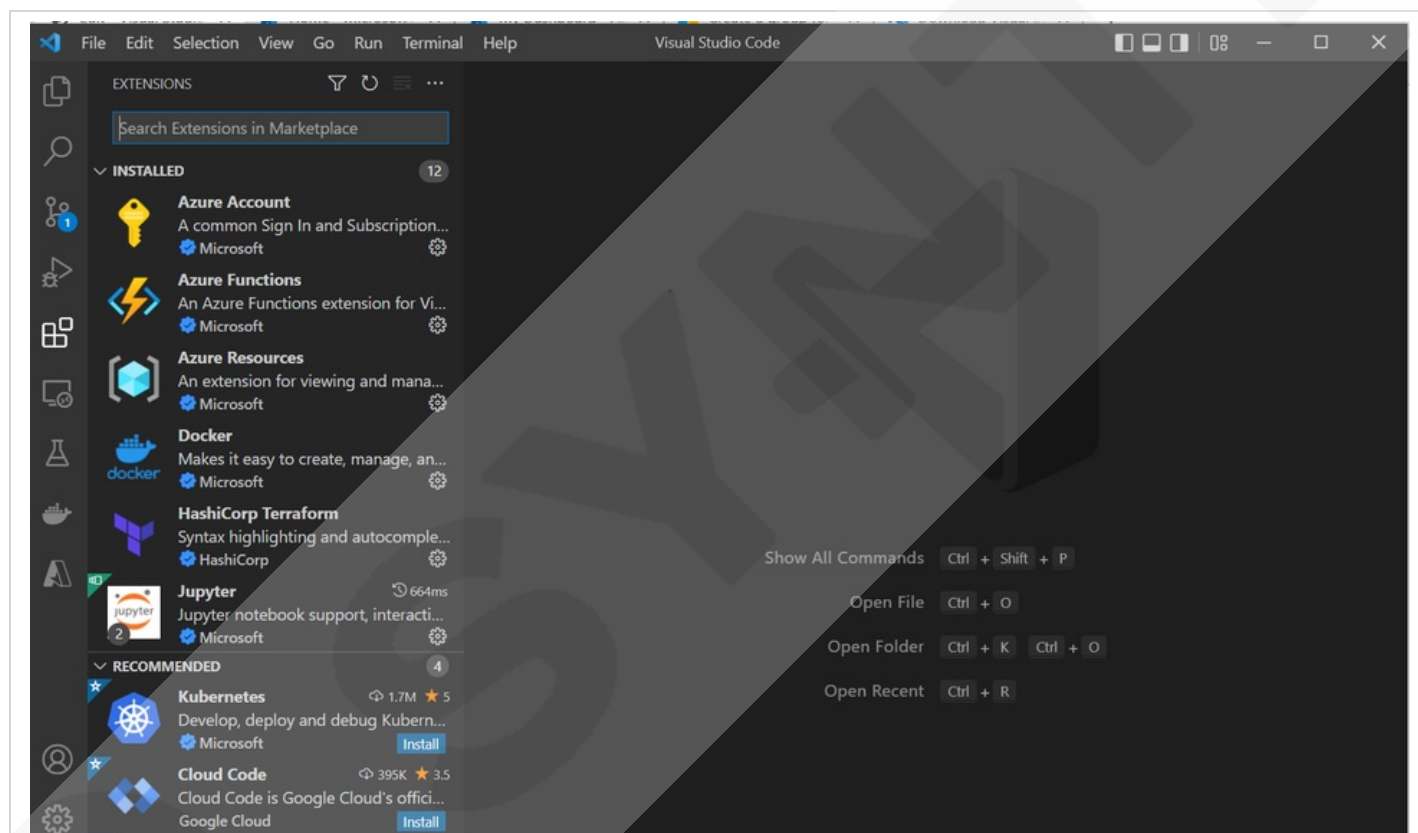
Dockerizacija aplikacija producera i consumera

- Napravite Dockerfile datoteke koje određuju bazne slike, instaliraju potrebne ovisnosti i kopiraju izvorni kod aplikacija u kontejnere
- Izgradite Docker image iz Dockerfile datoteka
- Pokrenite kontejnere iz Docker imagea pazeći pritom na kojoj mreži se nalaze

Container Apps

Potrebno je instalirati određene alate kako biste lakše izvršavali ovu laboratorijsku vježbu:

1. **Visual Studio Code** - ako ga već nemate, možete ga preuzeti na sljedećem linku: <https://code.visualstudio.com/download>
2. Nakon preuzimanja Visual Studio Codea unutar samog alata pronađite **extensions**.



Slika 10. Visual Studio Code extensions

Iz ovog izbornika odaberite i instalirajte **Azure Account** i **Azure Container Apps**.

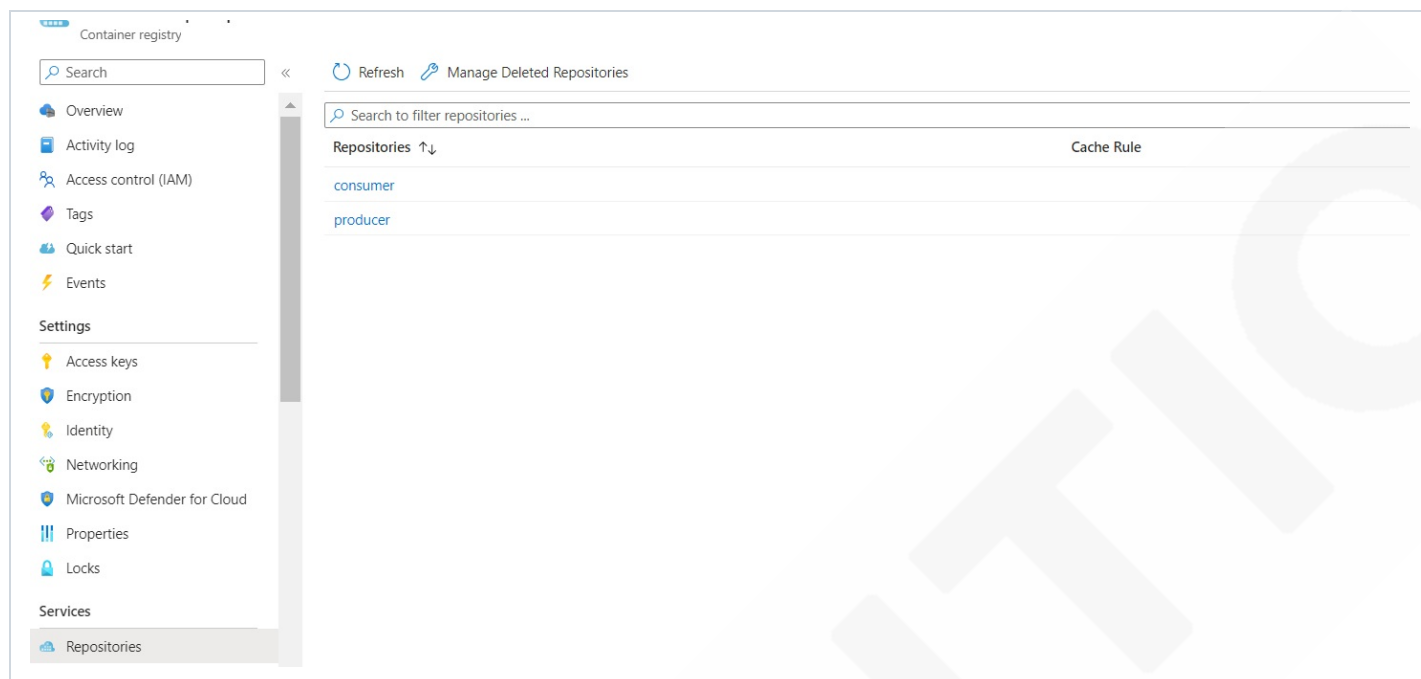
Note Možete koristiti i neki drugi uređivač koda, ali u tom slučaju na vama je da istražite kako koristiti Azure container apps unutar njih.

U ovoj dijelu laboratorijske vježbe ćete:

- koristiti Visual Studio Code za rad s Azure servisima
- stvoriti Container registry i pohraniti Docker image-a
- stvoriti Container Apps

Kreiranje Container registry-a

Nakon što ste izgradili Docker image potrebno ih je pohraniti na Azure unutar spremnika container registry-a. Azure Container Registry je usluga za izgradnju, pohranjivanje i upravljanje slikama spremnika. Za stvaranje Container Registry-a i pohranu izgrađenog Docker image-a slijedite upute s ovog linka. Na svom stvorenom Container Registry-u provjerite jesu li pohranjeni vaši Docker image-i.



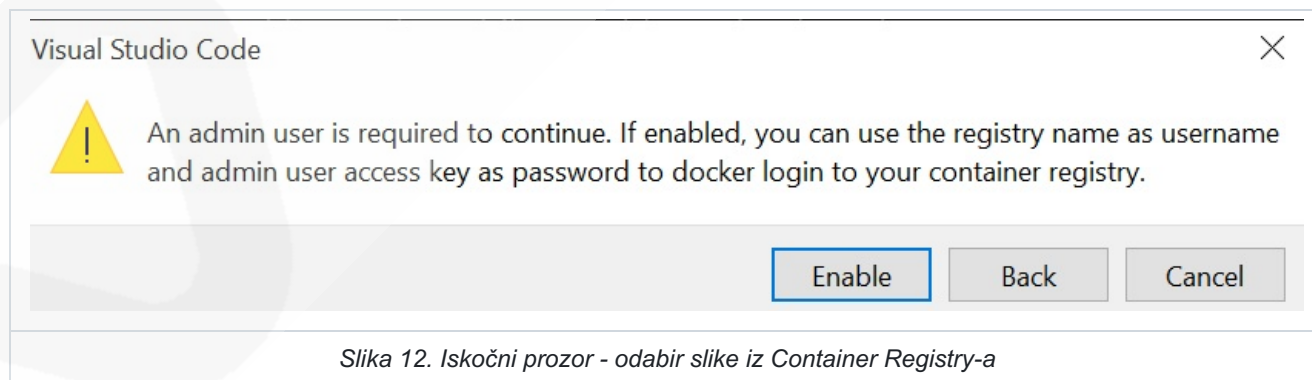
Slika 11. Sadržaj Container registry-a

Container Apps

Proširenje Azure Container Apps za VSCode omogućuje vam odabir postojećih resursa ili stvaranje novih Container Apps-a. Svaka Container App je dio Container Apps okruženja. Okruženje pruža izoliranu mrežu za jednu ili više aplikacija što im omogućuje jednostavno povezivanje. Prvo ćete kreirati okruženje, pa nakon toga Container App. Unutar VSCodea pritisnite F1 i unesite Azure Container Apps: Create Container Apps Environment. U nastavku trebate unijeti ime i regiju. Kao ime upišite fervjestina (ime resursne grupe). Azure počinje stvarati okruženje za vas, te također stvara i radni prostor za analizu logova. Ovo možete provjeriti unutar svoje resursne grupe na Azure portalu.

2. Zadatak

Sada kada imate stvoreno Container Apps okruženje potrebno je stvoriti dva Container Appsa s Docker image-ima producer i consumera koje ste pohranili unutar Container Registry-a. Postupak je sličan kao i prethodni korak samo trebate odabrati imagea iz prethodno stvorenog Container Registryja. Kod odabira slike iz Container Registry-a iskočit će vam prozor na slici te izaberite Enable kako bi dozvolili pristup image-ima.



Slika 12. Iskočni prozor - odabir slike iz Container Registry-a

Stvorene Container Appse trebali bi ste vidjeti u svojoj resursnoj grupi na Azure portalu.

Uvjerite se da u Event Hubu vidite incoming messages i da u consumer Container Appu pod karticom Log stream vidite primljene poruke.

Primjer log stream-a consumer Container App-a:

[illegible]

Slika 13. Log Stream primjer