# Povezava za prijavo seminarja na učilnici: <https://ucilnica.fri.uni-lj.si/mod/wiki/view.php?id=46993>

Formirajte skupine po dva (2) študenta in se prijavite na zgornjo povezavo.

# 1. Baza "employees" - več skupin, priporočamo študentom Upravne informatike

* Orodja - PowerDesigner, različni SUPB
  + MariaDB ali MySQL
  + PostgreSQL
  + SQLite (Python standardna knjižnica)
  + MS SQL Server
  + \*Oracle (namestitev je bolj zahtevna)
* Podatki:
  + Employees sample database: <https://dev.mysql.com/doc/employee/en/>  
    Zaradi velikosti podatkov morate uvoziti dump datoteko v lokalno MariaDB bazo na način:  
    mysql -h yourhostname -u username -p databasename < yoursqlfile.sql  
    kjer se *mysql.exe* nahaja v bin direktoriju namestitve MariaDB.  
    Dump datoteke popravite še tako, da v prvo vrstico napišete *use Employees*  
    najprej pa poženete *employees.sql*. Datoteke so na naslovu: <https://github.com/datacharmer/test_db>, pazite na vrstni red uvoza.  
    Shemo baze vidite na: <https://dev.mysql.com/doc/employee/en/sakila-structure.html>
* Naloge:
  + Uvoz v PB
  + Reverzni inženiring s PD
  + Ročni popravki konceptualne sheme in primerjava z "5 Employees Structure"
  + Dopolnitev konceptualne sheme z novimi gradniki po izbiri:
    - Dodajte beleženje dopustov in bolniških odsotnosti
    - Dodajte podatke o vzdrževalnih članih (otroci, partnerji)
    - Dodajte podatke o službenih vozilih in njihovi uporabi
  + Generiranje prenovljenega logičnega in fizičnega modela, ter SQL skripte
  + Implementacija in polnjenje prenovljene baze (tudi novih gradnikov, ekstrakcija ali generiranje)
  + Definicija 5 smiselnih transakcij
  + Izvedba 5 smiselnih netrivialnih poizvedb
  + Izvedba 5 smiselnih netrivialnih dostopov iz Pythona in povezanih poizvedb
  + Analiza performans  in implementacija dodatnih indeksov

# 2. Odprti podatki z ML/DM/AI ali DB/DW/DL analitičnimi orodji

Izberite si kombinacijo SUPB, analitičnega orodja in podatkov. Raziščite zmožnosti izbranega orodja, podporo korakom CRISP-DM, integracijo z izbranim SUPB in različne načine uporabe. Vse skupaj ilustrirajte z uporabo na večji podatkovni zbirki (vsaj 2 milijona zapisov).

* Orodja:
  + ML/DM/AI:
    - MindsDB: <https://mindsdb.com>
    - Rapid Miner: <https://rapidminer.com/>
    - Knime: <https://www.knime.com/>
    - Orange: <https://orangedatamining.com/>
    - Weka: <https://www.cs.waikato.ac.nz/ml/weka/>
    - R/Rattle: <https://rattle.togaware.com/>
    - Python/Scikit-learn + GUI (če kdo kaj najde)
    - Python/Scikit-learn/Ipython widgets
    - Python/Scikit-learn iz Excela
    - DataMelt: <https://datamelt.org/>
    - H2O (open source): <https://www.h2o.ai/products/h2o/>
    - ELKI: <https://elki-project.github.io/weka>
    - Tableau: <https://www.tableau.com/academic/students>
    - ... lastna izbira (po dogovoru)
  + DB/DW/DL:
    - MinIO: <https://min.io>
    - Microsoft SQL Server Data Tools
    - CRISP-DM: <https://en.wikipedia.org/wiki/Cross-industry_standard_process_for_data_mining>
* Podatki:
  + Poiščite primerno velike podatke med viri odprtih podatkov, npr.: <https://www.bernardmarr.com/default.asp?contentID=960>

# 3. Povezovanje COVID-19 podatkovnih zbirk (ena skupina)

Preglejte različne vire podatkovnih zbirk za COVID-19. Preglejte sheme zbirk (strukturo tabel) in identificirajte atribute, preko katerih bi lahko povezovali podatke (npr. časovno in lokacijsko, po ključnih besedah, …). Iz pregledanih zbirk izberite vsaj 10 povezljivih, jih implementirajte v izbrano podatkovno bazo, izvedite polnjenje in korake, potrebne za povezavo (kreiranje novih atributov, definicija tujih ključev, …) in realizirajte vsaj 10 netrivialnih poizvedb nad povezanimi tabelami. Polnjenje in transformacijski koraki naj bodo realizirani tako, da bodo omogočali sprotno polnjenje tabele skozi čas. Na koncu podatke analizirajte in vizualizirajte z orodji iz točke 2.

Viri podatkov:

* Amazon: <https://aws.amazon.com/covid-19-data-lake/>
* EU: <https://data.europa.eu/euodp/en/data/dataset/covid-19-coronavirus-data>
* Nature: <https://www.nature.com/collections/ebaiehhfhg>
* Kaggle: <https://www.kaggle.com/allen-institute-for-ai/CORD-19-research-challenge>
* Google: iščite COVID-19 open data

# Normalizacija in uporaba podatkovne tabele »Smetarine« (lahko več skupin uporabi te podatke, vendar z različnimi izbranimi orodji)

Gre za podatke o obračunavanju smetarin komunalnega podjetja za leti 2019 in 2020. Tabela ni normalizirana. Najprej prenesete podatke na svoj računalnik v MySQL bazo. Tabela »smetarine« se nahaja na pb.fri.uni-lj.si v shemi tup. Prenos lahko izvedete na več načinov (izvoz v datoteko, preko ODBC-ja ali PyODBC-ja). Izberite svojega in ga v nalogi opišite.

Podatki za normalizacijo:  
smetarine(StInkasa,DatumInkasa,InkasoOpis,stevilka,LastnikL,PlacnikL,HisnaStevilka,HisnaPoddelilka,DavcniZavezanec,ProracPorab,SkZnesek,SkDavek,SkZaPlacilo,DatumDokumenta,DatumValuta,Izravnava,Storno,ZamObresti,GKSaldo,ArtikelL,DavcnaSkupinaL,DavekOpis,Procent,Kolicina,ProdajnaCena,ProdajnaCenaZDavkom,StClanov,TagX,ObjektL,MerskaEnotaL,Enota,EnotaOpis)

Funkcionalne odvisnosti:

StInkasa->DatumInkasa,InkasoOpis

stevilka-> StInkasa,LastnikL,PlacnikL,SkZnesek,SkDavek,SkZaPlacilo,DatumDokumenta,DatumValuta, Izravnava,Storno,ZamObresti,GKSaldo

ali velja odvisnost stevilka,ArtikelL->Kolicina ?

PlacnikL->HisnaStevilka,HisnaPoddelilka,DavcniZavezanec,ProracPorab

ArtikelL->DavcnaSkupinaL,ProdajnaCena,ProdajnaCenaZDavkom

DavcnaSkupinaL->DavekOpis,Procent

LastnikL,ObjektL->StClanov

MerskaEnotaL->Enota,EnotaOpis

TagX kljukica označuje prvo postavko na računu. Na računu je lahko več objektov in postavke za nov objekt se začnejo s kljukico TagX in novo številko objekta

primer je račun št. 1836508

opisi atributov, ki niso jasni sami po sebi:

inkaso je skupna oznaka za mesečni obracun računov

Izravnava - stotinska izravnava, da ima znesek dve decimalki

Storno - dan ko je bil račun storniran

ZamObresti - koliko je bilo na računu pripisanih zamudnih obresti do dneva inkasa

GKSaldo - koliko ima plačnik stanje v glavni knjigi na dan inkasa

še način izračuna vrednosti:

SkZnesek = vsota vrednosti postavk po formuli: Kolicina \* ProdajnaCena

SkDavek = vsota vrednosti postavk po formuli: Kolicina \* ProdajnaCena \* Procent / 100

SkZaPlacilo = vsota vrednosti postavk po formuli: SkZnesek + SkDavek + Izravnava

Tako normalizirane podatke uporabite za testiranje različnih orodij, navedenih v tem dokumentu, in realizirajte vsaj 10 netrivialnih poizvedb nad povezanimi tabelami.

Viri podatkov:

* Smetarine: <http://pb.fri.uni-lj.si/phpmyadmin/sql.php?server=1&db=tup&table=smetarine&pos=0&token=54e54d33e247df19eb3d4c9b6f478dee>