Prva domaća zadaća iz Podatkovnog inženjerstva

ETL & Event processing

Prerequisites for the first lab assignement

before the lab assignement you should install:

- Python (>=3.9,<3.11)
- Poetry

Poetry is a tool for dependency management and packaging in Python. It allows you to declare the libraries your project depends on and it will manage (install/update) them for you. The following script should be run to install it

```
(Invoke-WebRequest -Uri https://install.python-poetry.org -UseBasicParsing).Content | py -
```

Poetry environment is defined in the pyproject.toml file. To generate the environment run:

```
poetry update
```

This command should generate a virtual environment and install all the dependencies defined in the pyproject.toml file. The state after running this command is saved in poetry.lock file and all the dependencies are installed in the .venv folder. Poetry lock file is a cool concept since you can share it with your team and everyone will have the same versions of the dependencies installed. For somebody else who wants to replicate your environment they only need to run the following command:

```
poetry install
```

Poetry install is simmilar to poetry update, but it takes the whole environment from the lock file and installs it, it does not take any later dependencies or change the lock file.

Prefect basics

Documentation at https://docs.prefect.io/ Prefect is a workflow management system that allows you to define and execute workflows in Python. It comes prepackaged with a web ui called Prefect Orion. There are couple specific Prefect concepts which are benefitial to know:

Prefect Concept	Description
Flows	Flow is a Python function which is runnable within Prefect, it comes anotated with the @flow anotation
Flow Runs	Flow run is an instance of the flow which has been registered with some parameters and can be in multiple statuses, either Scheduled to be run in future, being run or finished in one of couple states
Deployment	Deployments are enriched flow instances which contain information on where Flows should run (which kind of agents (specified through pools)) and can have some schedules appended as well (e.g. via cron)
Work Pools	Work pools as a concept is similar to message queues and they specify which flows will be taken by which agent. This allows the writer of flows to setup different environments for different usecases.
Blocks	Blocks hold parameters and code environments which are taken by workers. There are a lot of implementations of prefect block, but the one which will be used in this lab assignement is the local_file block.

Prefect orion UI

To view results from prefect runs, schedule new runs, view logs, etc. we will be using the Prefect Orion UI. To start the UI run the following command:

```
poetry shell
prefect server start
```

This command should start up a local web interface with default URL http://127.0.0.1:4200, and also a prefect api url which will be connection point for the prefect agents later on.

Prefect deployments

Prefect deployments can be run in multiple kinds of environments, spanning from the server itself, workers, docker containers, kubernetes clusters, etc. Prefect deployments are created with a specific needed set of metadata, name, path to it, work queue name, work pool name, work pool name and function pointer.

```
from prefect import flow
from prefect.deployments import Deployment
import os

@flow
def flow_code():
    print("I am a flow :)")

deployment_instance = Deployment.build_from_flow(name="flow_name",
                                                 path=os.path.abspath(os.path.curdir),
                                                 work_queue_name="demo",
                                                 flow=flow_code,
                                                 work_pool_name="demo"
                                                )
deployment_instance.apply()
```

Spinning up agents

In this lab assignment local prefect agents will be used. After the deployment has been created we need something to run it on. It is time to spin up some workers. To run the workers go inside of poetry shell, set the api where the agent will be listening and run the agent itself:

```
poetry shell
export PREFECT__SERVER__API=http://localhost:4200/api
prefect agent start --pool demo
```