

Oppimispäiväkirja  
5G00EU62-3005

Kuusisto Jaakko

Tietotekniikan tutkinto-ohjelma

Ohjelmistotekniikka

Sisällysluettelo

[1 Tehtävä 1 3](#_Toc188880146)

[2 Tehtävä 2 4](#_Toc188880147)

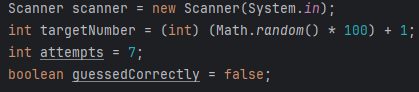
[3 Tehtävä 3 6](#_Toc188880148)

[4 LÄHTEET 14](#_Toc188880149)

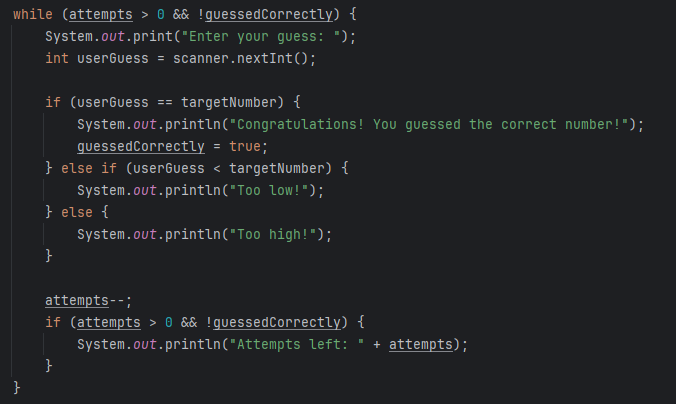
# Tehtävä 1

Tehdään yksinkertainen arvauspeli, jossa aluksi alustetaan muuttujat ja loput ohjelmasta pyörii komentorivi while() -silmukassa.

Numeroiden alustus ja satunnaisluku math.random() -funktiolla:



Alustuksen jälkeen silmukka, joka päättyy yritysten loppuun tai oikeaan arvaukseen.



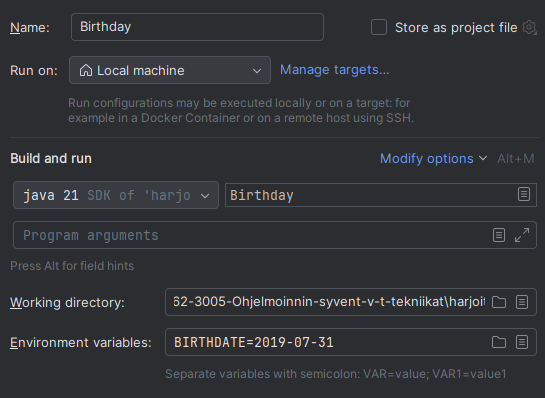
Linkki projektin lähdekoodiin osoitteessa:

<https://github.com/MikeDanton/5G00EU62-3005-Ohjelmoinnin-syvent-v-t-tekniikat/tree/main/harjoitus1/untitled/src>

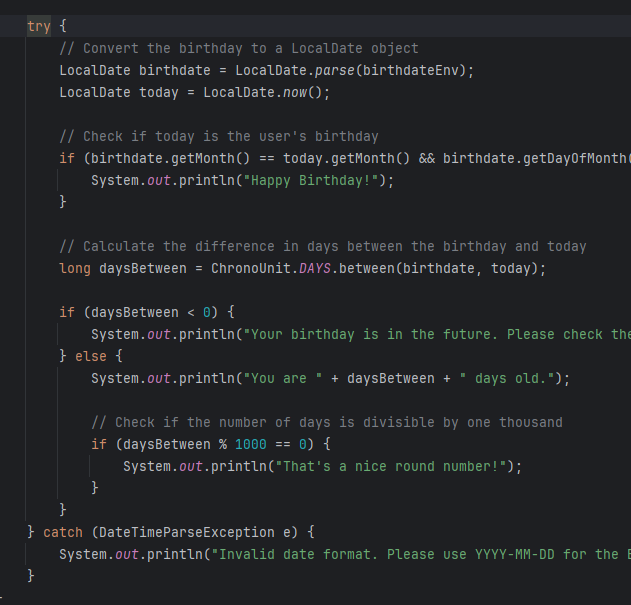
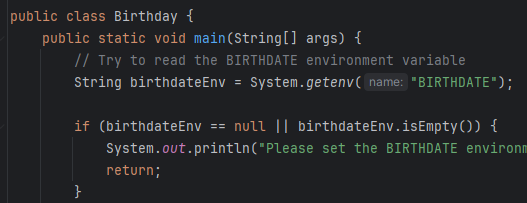
# Tehtävä 2

Tehtiin Birthday-ohjelma, joka tulostaa syötteitä luetun BIRTHDATE nimisen ympäristömuuttujan perusteella.

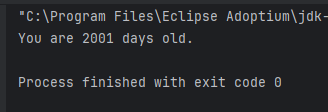
Aluksi luotiin ympäristö ja asetettiin muuttuja ideen:



Aluksi asetetaan paikalliset muuttujat ja tehdään null-tarkistus BIRTHDATE:lle.  
  
  
 Tämän jälkeen ohjelma on pitkä lista if-lauseita, jotka käyvät läpi ohjelman vaadittuja toiminnallisuuksia. Ohjelma on try-catch -blokin sisällä, jos DateTimeParseException heittää virheen väärästä tallennusformaatista jäsentelyn aikana.



Ohjelma käyttää ChronoUnit ja LocalDate kirjastoja ajan käsittelyyn ja esim. jakojäännöstä jaollisuuden tarkistamiseen.



Yllä asetettu syntymäpäivä pitäisi antaa 2000, mutta voi olla, että LocalDate ja ChronoUnit eivät ota aikavyöhykkeitä huomioon täysin sellaisinaan tms. Bugi vaikuttaa olevan tämän tehtävänannon ulkopuolella.

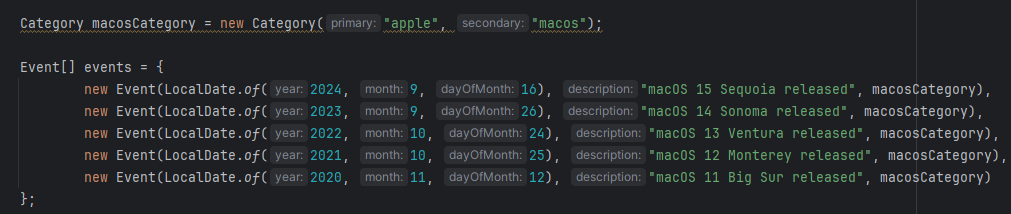
Linkki lähdekoodiin:

[5G00EU62-3005-Ohjelmoinnin-syvent-v-t-tekniikat/harjoitus2 at main · MikeDanton/5G00EU62-3005-Ohjelmoinnin-syvent-v-t-tekniikat](https://github.com/MikeDanton/5G00EU62-3005-Ohjelmoinnin-syvent-v-t-tekniikat/tree/main/harjoitus2)

# Tehtävä 3

Tehtävänä oli tehdä ohjelma, joka käytti kurssireposta löytyviä Event- ja Category-luokkia.

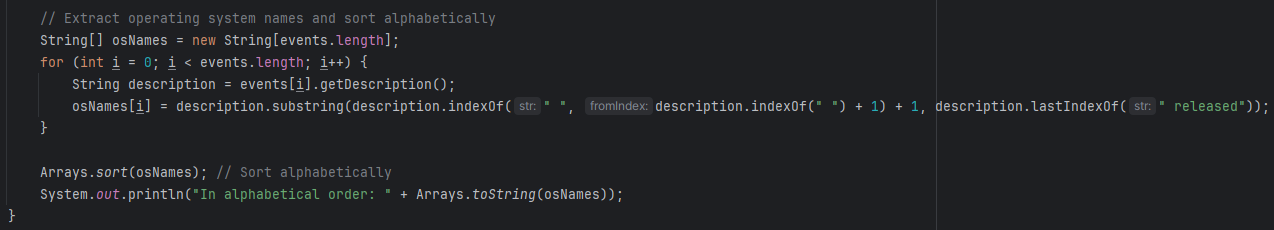
Aluksi muuttujien alustus käytettyjen luokkien rakentajien ja tehtävänannon datan perusteella:



For-silmukassa tehdään tulostuksen käsittely. LocalDate tulostaa isoilla kirjaimilla muuttujan, joten tarvitaan ylimääräistä käsittelyä tehtävänannon tulostukseen.



Seuraavaksi kerätään käyttöjärjestelmien nimet String-taulukkoon, järjestetään se Arrays.sort()-metodilla ja tulostetaan.



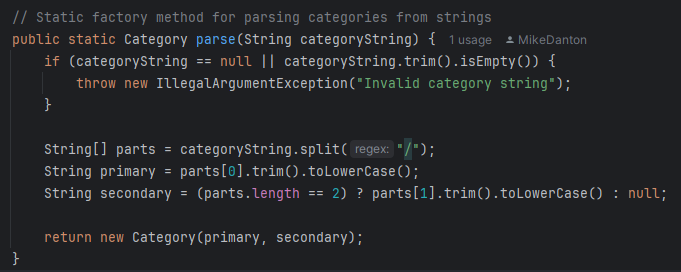
Linkki lähdekoodiin:

<https://github.com/MikeDanton/5G00EU62-3005-Ohjelmoinnin-syvent-v-t-tekniikat/tree/main/harjoitus3>

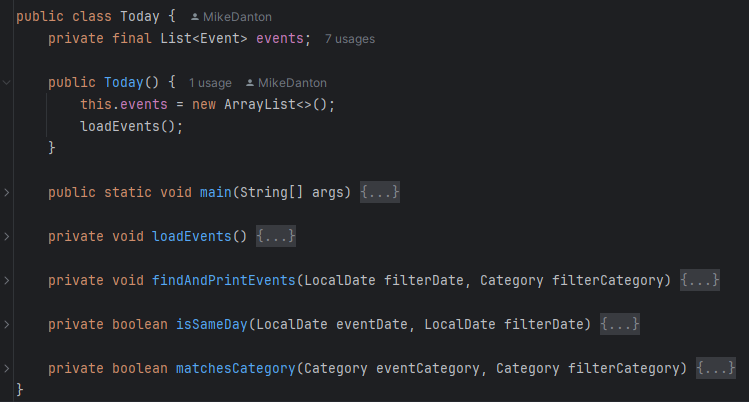
# Tehtävä 4

Tehtiin Today-ohjelma, johon voi lisätä/tarkastella päivän tapahtumia komentoriviargumenttien perusteella. Tehtävään annettiin valmiina Category- ja Event-luokat. Lisäksi valmiina oli pohja Today-luokka, jota tuli muokata tehtävänantoa vastaavaksi.

Category-luokkaan lisättiin tehdasmetodi, joka lajitteli String-objektin ja palautti uuden Category-objektin. Kaikki isot kirjaimet muutettiin pieniksi käsittelyn helpottamiseksi.



Alla yhteenveto Today-luokan toiminnasta



loadEvents(), metodi esimerkkidatan alustukseen rakentajassa.

findAndPrintEvents(), metodi tapahtumien tulostukseen päivän ja kategorian perusteella. findAndPrintEvents () käyttää isSameDay() metodia sisäisesti, joka on apumetodi tarkastamaan, onko päivä sama.

matchesCategory() on myös apumetodi, jota findAndPrintEvents() käyttää.

Pääohjelman toiminta:

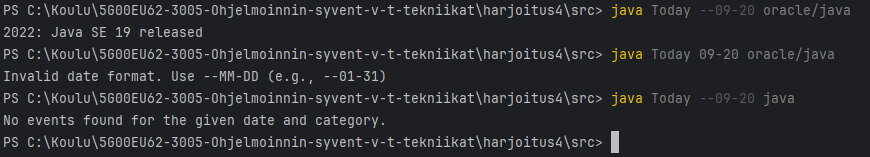


Poimitaan aluksi komentoriviargumentit omiin String-objekteihinsa ja tehdään virheenkäsittely.

Tämän jälkeen muutetaan input-String -objektit localDate- ja Category-objekteiksi.

Lopuksi alustetaan uusi Today-objekti esimerkkidatalla ja kutsutaan findAndPrintEvents() argumentteina localDate- ja Category-objektit ja annetaan ohjelman tulostaa syöte komentoriville.

Esimerkkitulosteita:



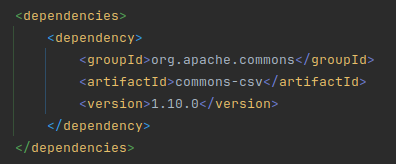
Linkki lähdekoodiin:

<https://github.com/MikeDanton/5G00EU62-3005-Ohjelmoinnin-syvent-v-t-tekniikat/tree/main/harjoitus4/src>

# Tehtävä 5

Oppimistavoitteina tällä viikolla oli ainakin ulkoisten kirjastojen käyttö, standardoitujen merkkijonojen käsittelly, sekä hakemistojen käyttö OS:ssä.

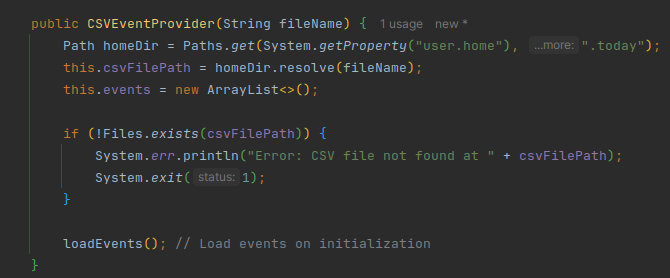
Ympäristöksi valittiin Maven, koska AI kertoi siihen kirjastojen lisäämisen olevan helpointa ja generoi seuraavan:



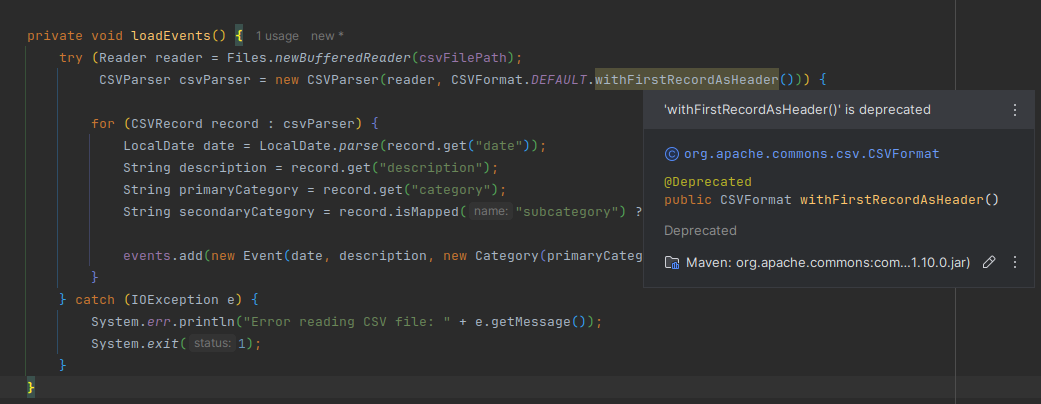
Idenä toimi intelliJ IDEA ja projekti saatiin toimimaan.

Standardoiduista merkkijonoista .csv ei tue sisäkkäisyyttä, (=nesting) (ilman kikkoja) joten käsittely on melko simppeliä. Huomautuksena kaikki toimivat kuitenkin samalla tavalla, kirjasto kerää jonkin taulukon kirjaston omaa luokkaa käyttäviä objekteja ja nämä pitää muuttaa haluttuun muotoon ja jatkokäsitellä. Täten, tekoälyn annettiin hoitaa ongelma, johon menisi muuten turhaa aikaa perehtyä kirjaston metodeihin jms. ”Comma separated values” on yksinkertainen, mutta näissäkin voi olla eksoottisempia syntakseja.

Rakentajassa asetettiin polku ja tarkastettiin onko tiedosto olemassa. AI:n mukaan tämä Javan tapa asettaa kotihakemisto on ”crossplatform” ja ainakin Linuxissa tämä toimi.



Tämän jälkeen luettiin jäsentelyn lähteestä (record-termi yleinen), jotka saatiin sijoitettua suoraan Stringiin. Ide näyttää Deprecated -varoitusta, joten varmasti uudempi tapa olisi olemassa, mutta tämäkin toimi.



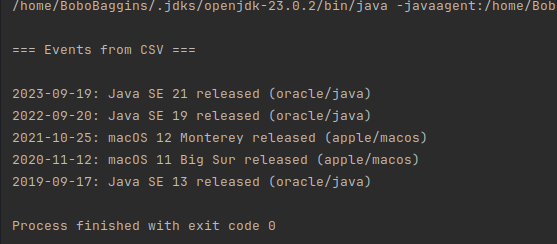
GitHubista löytyi Interface-tiedosto joten puskettiin haluttu data sen läpi:



Ja pääohjelmassa tulostettiin tiedot:



Esimerkkitulostus:



Linkki lähdekoodiin:

<https://github.com/MikeDanton/5G00EU62-3005-Ohjelmoinnin-syvent-v-t-tekniikat/tree/main/harjoitus5/src/main/java>

# Tehtävä 6

Tehtävänä oli toteuttaa EventManager-luokan metodit addEventProvider ja removeEventProvider, niin että testiohjelma Today toimii ilman UnsupportedOperationException-poikkeuksia, ja samaa "event provideria" eli tapahtumatuottajaa ei lisätä kahteen kertaan. Oppimistavoitteena oli ohjelman toimiminen oikein, koodissa on ratkaisua selittäviä kommentteja, toteutus on kuvattu oppimispäiväkirjassa tekoälyn käyttö, että siitäkin voisi oppia edes jotain.

Pohjaksi otettiin edellisen viikon tehtävä, mutta Lisättiin interfaceen getter String getIdentifier(). Githubista kopioitiin EvenManader ja EventProvider 1 ja 2.

Pääohjelmaan tehtiin testitulosteita ja käytettiin valmiiksi tehtyä EventManagerin singleton –suunnittelumallia.



Oletettiin, että edellisillä viikoilla mainittu omien ratkaisujen käyttö on sallittua. Joten EvenManageri tehtiin käyttämään hashmap-säiliötä, mutta lisättiin myös arraylist-ratkaisu mukaan.

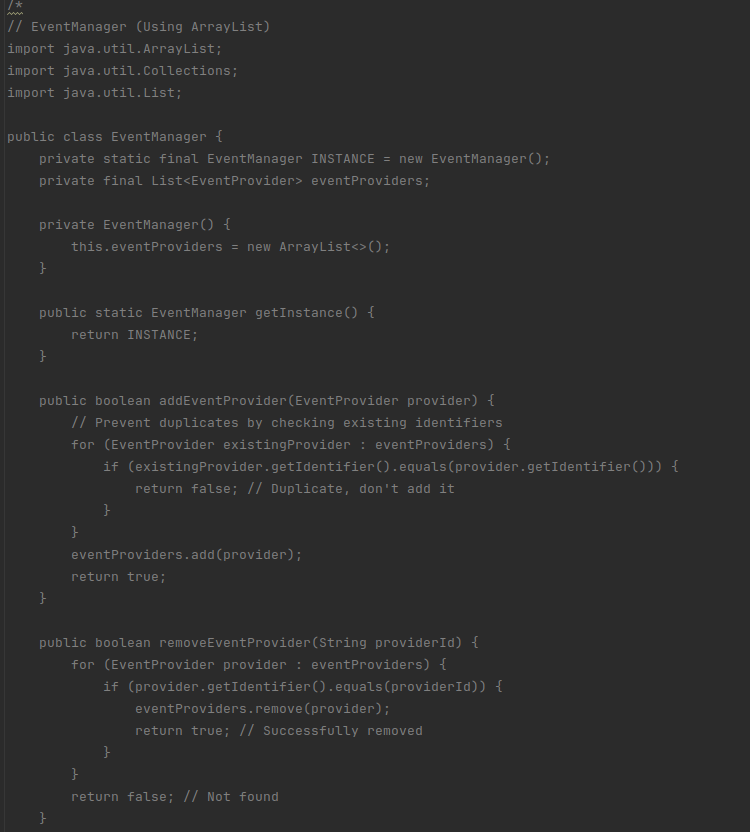


Perusteena ratkaisulle puhdas ”Because AI told me so”.

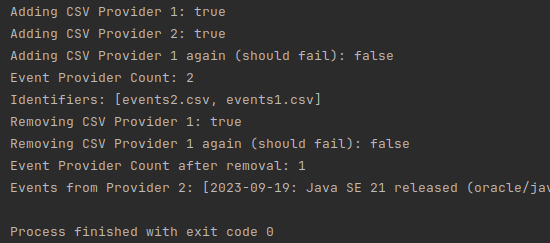


Tämäkin selitys uskottiin sellaisenaan.

Ohjelman käyttötarkoituksesta ei voi tehdä suurempia johtopäätöksiä, mutta mapin avainpari ratkaisee automaattisesti uniikkivaatimuksen ja O(1) hakuaika voi olla myös hyödyllinen. Arraylla kirjoitettaisiin if-lauseita sekaan uniikkivaatimuksesta varmistuttaessa.



Pääohjelman toimimisesta varmistuva tulostus:



# LÄHTEET

https://chatgpt.com/

Maven: org.apache.commons:commons-csv:1.10.0