

Bestudeer de video van de volgende les geheel:

Klassen en objecten: inkapseling, overerving en veelvormigheid

Experimenteer zelfstandig met de daarin behandelde voorbeeld-programma's. Een link naar de broncode daarvan staat onder de video. In de volgende les kun je hierover vragen stellen.

Opgaven bij de huidige les:

In te leveren uiterlijk 2 dagen vóór de volgende les.

Niveau 1

Maak een programma dat aan de gebruiker vraagt twee lijsten van namen in te voeren. Geef het programma een keuzemenu (gewoon in de console) waarmee de gebruiker kan kiezen uit 4 mogelijkheden:

1. Geef alle namen die in tenminste één van beide lijsten voorkomen, gesorteerd op alfabet, met doublures verwijderd. Dit heet de vereniging (union) van de twee lijsten.
2. Geef alle namen die wel in de eerste lijst voorkomen, maar niet in de tweede. Dit heet het verschil (difference) van de eerste en de tweede lijst.
3. Geef alle namen die wel in de tweede lijst voorkomen maar niet in de eerste. Dit heet het verschil (difference) van de tweede en de eerste lijst.
4. Geef alle namen die in beide lijsten voorkomen. Dit heet de doorsnede (intersection) van de twee lijsten.

Schrijf met dit doel 3 functies, functie *getUnion* voor mogelijkheid 1, functie *getDifference* voor mogelijkheid 2 en 3 en functie *getIntersection* voor mogelijkheid 4. Vermijd rechtstreeks gebruik IN een bepaalde functie, van variabelen die BUITEN die functie gedefinieerd zijn.

Niveau 2

Zoek uit hoe je teksten kunt wegschrijven naar en teruglezen uit het achtergrondgeheugen van je computer. Hint: Zoek in de officiële Python documentatie naar de *open* functie en lees daar nauwkeurig hoe je tekstbestanden kunt wegschrijven en teruglezen.

Maak een programma dat hetzelfde doet als het programma van niveau 1, maar nu met lijsten die in het achtergrondgeheugen (ook wel, al dan niet terecht, harde schijf genoemd) staan opgeslagen. Het resultaat wordt als een bestand met een passende naam weggeschreven in het achtergrondgeheugen, waarbij de oorspronkelijke bestanden intact worden gelaten. De resultaten van mogelijkheid 1, 2, 3, en 4 hebben andere bestandsnamen, die afgeleid zijn uit de namen van de invoerbestanden.

Niveau 3

Maak een programma dat hetzelfde doet als het programma bij niveau 2, maar dan op data die oorspronkelijk afkomstig uit MS-Teams bij Manage Team (via de ... in het linker panel).

Cut de deelnemerlijst en paste hem in je editor. Save vervolgens je bestand.

Het programma dient de namen er uit te filteren en als lijst weg te kunnen schrijven. Let op: Elke regel dient ALLEEN de naam van een persoon te bevatten. Bedenk hoe je deze regels kunt herkennen en uitfilteren.

Zet de personen op alfabet. Maak twee lijsten, eentje op voornaam en eentje op achternaam. Bedenk hoe je dit voor elkaar kunt krijgen.

Voor het overige dient het programma op de eenmaal naar schijf geschreven lijsten weer de functies *union*, *difference* en *intersection* te kunnen uitvoeren en de resultaten onder zinnige namen naar schijf te kunnen schrijven.

Extra: Zorg er dan voor dat de stap via de editor achterwege kan blijven, dus dat je rechtstreeks in Python kunt pasten. Dit heeft nog al wat voeten in de aarde, maar staat netjes beschreven op www.StackOverflow.com. Bedenk zelf wat goede zoektermen zijn om dit antwoord te vinden.

Opmerking: Om kunnen gaan met StackOverflow is essentieel voor iedere programmeur die in z'n beroeps-praktijk iets anders wil doen dan voortdurend hetzelfde kunstje.

Extra 2: Als je de smaak nu helemaal te pakken hebt, geef je programma dan een tkinter GUI.

--//--