

Eindopdracht

Leerlijn Programming Fundamentals (15 EC)

V-1.1



Inhoudsopgave

EINDOPDRACHT PROGRAMMING FUNDAMENTALS	3
Integrale eindopdracht	3
Algemene opdrachtbeschrijving	3
Op te leveren producten.....	3
DEELOPDRACHTEN.....	4
Deelopdracht 1. Ideefase	4
Deelopdracht 2. Applicatie	5
Deelopdracht 3. Verantwoordingsdocument	6
QUICKSCAN	7
BEOORDELINGSCRITERIA.....	7

Eindopdracht Programming Fundamentals

Integrale eindopdracht

De leerlijn Programming Fundamentals behandelt de belangrijkste basisprincipes van het programmeren. Om deze leerlijn af te ronden dien je de volgende leeruitkomst aan te tonen:

Programming Fundamentals

De student past belangrijke basisprincipes van het programmeren toe in een zelfgeschreven applicatie.

Algemene opdrachtbeschrijving

In ons dagelijks leven vertrouwen we steeds meer op applicaties om ons te informeren, te organiseren en ons te verbinden met de wereld om ons heen. Deze digitale tools versimpelen complexe taken, bieden directe toegang tot informatie en verhogen onze productiviteit aanzienlijk. Van simpele agenda's tot aan complexe analysesystemen, applicaties vormen de ruggengraat van moderne technologie die we elke dag gebruiken.

In deze opdracht ga je aan de slag met het ontwikkelen van je eigen Python applicatie. Python, bekend om zijn eenvoud en veelzijdigheid, is de perfecte taal om je ontwikkelvaardigheden mee te verbeteren. Je gaat eerst aan de slag met het beschrijven van jouw idee waarin je uitwerkt hoe je te werk zal gaan. Vervolgens ga je een command-line applicatie creëren die gegevens ophaalt en bewerkt met behulp van externe API's, waarbij je laat zien hoe je gebruikersinteractie mogelijk maakt en data op een zinvolle manier presenteert. Hiervoor documenteer je je code op een professionele manier en tijdens het maken van de deelopdrachten verantwoord je de gemaakte keuzes in een kort verslag.

Op te leveren producten

- Beschrijving ideefase
- Applicatie
- Verantwoordingsdocument

Deelopdrachten

In de eerste deelopdracht tref je de voorbereidingen om je applicatie te gaan bouwen. In de tweede deelopdracht programmeer je de applicatie. Afsluitend schrijf je voor de derde deelopdracht het projectverslag, waarin je onder andere het project evalueert en de door jou gemaakte keuzes bespreekt.

Deelopdracht 1. Ideefase

Voor je kan beginnen met het bouwen van de command-line applicatie is het belangrijk om goed in kaart te brengen wat je precies wilt gaan doen en hoe je dat wilt gaan doen. Hiervoor is het van belang dat je over de volgende onderdelen gaat nadenken en deze opneemt in je ideefase:

- Wat voor applicatie je wilt bouwen;
- Welke 3 verschillende en passende functionaliteiten de applicatie heeft;
- Welke 2 endpoints van welke API(s) je hiervoor gaat gebruiken en waarom.

Om je een beeld te geven van wat voor soort applicatie je zou kunnen ontwikkelen volgt hier een voorbeeld van een ideefase:

Een wereldwijde weerapplicatie

Voor de eindopdracht van Programming Fundamentals ga ik een applicatie bouwen waarbij gebruikers informatie kunnen opzoeken over verschillende hoofdsteden van landen en daarbij het huidige weer. De applicatie handelt ook fouten af en geeft duidelijk aan wanneer gebruikers onjuiste input geven. Om herhaalde code te voorkomen, zijn verschillende helper functies gemaakt.

1. De applicatie toont een aantal landen op basis van ingevoerde gegevens. Gebruikers kunnen filteren op aangrenzende landen, continent en populatie. Door input in de terminal haalt de applicatie data op.
2. De weergave van data is vereenvoudigd, zodat gebruikers in één oogopslag kunnen zien welke landen aan de wensen voldoen.
3. De applicatie toont het actuele weer op basis van de geselecteerde input van de gebruiker.

Ik ga de volgende twee endpoints gebruiken:

1. Voor het ophalen van de informatie over de hoofdsteden gebruik ik het volgende endpoint:
`https://restcountries.eu/rest/v2/capital/{capital}`
2. Voor het ophalen van het huidige weer van een gezochte hoofdstad, gebruik ik het volgende endpoint:
`https://api.openweathermap.org/data/3.0/onecall?lat=33.44&lon=-94.04&exclude=hourly,daily&appid={API key}`

Op te leveren:

- Beschrijving ideefase

Deelopdracht 2. Applicatie

In de eerste deelopdracht van je project heb je een ideefase beschreven, waarin je hebt opgeschreven wat voor soort applicatie je gaat bouwen, welke functionaliteiten de applicatie heeft en welke twee API's je gaat gebruiken ter ondersteuning van je applicatie. In de tweede deelopdracht gebruik je dit idee als leidraad om de applicatie te bouwen in Python. Nu ga je beginnen met het echte (en het leukste!) werk.

Je gaat tijdens deelopdracht twee daadwerkelijk aan de slag met het ontwikkelen van jouw command-line applicatie. Let op! Je applicatie moet wel voldoen aan de volgende voorwaarden:

- Er is een menustructuur aanwezig met meerdere lagen, waarmee de gebruiker minimaal 3 verschillende passende acties kan uitvoeren. Denk aan het opvragen van data, filteren en sorteren van informatie, doorlopen van beslissingsstructuren om tot een uitkomst te komen, etc.
- De acties zijn gebaseerd op de 3 verschillende functionaliteiten zoals beschreven in de ideefase;
- De gebruikte informatie wordt opgehaald door middel van minimaal 2 API-calls naar 2 verschillende endpoints (dit mogen ook twee verschillende API's zijn);
- Foute invoer door de gebruiker wordt afgehandeld door de gebruiker hier op de juiste manier over te informeren en terug te sturen in het keuzeproces (menu);
- Je handelt in lijn met best practices, om je code efficiënt en netjes te structureren.

Op te leveren:

- De code in een ZIP-bestand

Deelopdracht 3. Verantwoordingsdocument

Tijdens het programmeren heb je vast gemerkt dat er een hoop mis of goed kan gaan. Dit verleent zich tot mooie leermomenten voor jou als student! Misschien heb je wel je best gedaan om best practices te gebruiken zoals DRY, of liep je tegen een lastig code probleem aan, waar je veel moeite en uren aan hebt moeten besteden om dit op te lossen. Voor deelopdracht drie ga je je proces tijdens het programmeren omschrijven en verantwoorden.

In het verantwoordingsdocument werk je de volgende zaken uit:

- Je reflecteert op twee problemen die je bent tegengekomen tijdens de implementatie, en je legt uit hoe je deze uitdaging hebt aangepakt en opgelost.
- Je benoemt twee doorontwikkelingen waarop je code verbeterd zou kunnen worden en beargumenteert waarom dit wenselijk is.
- Je onderbouwt welke best practices je hebt terug laten komen in je code en hoe.

Op te leveren:

- Verantwoordingsdocument

Quickscan

Hieronder vind je een aantal randvoorwaarden waaraan de eindopdracht moet voldoen. De beoordelaar kan op basis van deze eisen de eindopdracht teruggeven en zal deze niet verder nakijken tot aan de eisen zijn voldaan.

Gebruik de quickscan om te zien of je voldoet aan de inleververeisen.

Algemene eisen:

- ☐ Documentatie (ideefase + verantwoordingsdocument) wordt ingeleverd als pdf, los van de broncode.
- ☐ De broncode wordt aangeleverd d.m.v. een ZIP-bestand van maximaal 50 MB.
- ☐ De eindopdracht is goed leesbaar zonder storende aanwezigheid van grammatica- en spellingsfouten.
- ☐ De omvang van de deelopdracht 1 en deelopdracht 3 beslaat maximaal 2.000 woorden.
- ☐ De broncode wordt ingeleverd zonder '.env-map' en '.idea'-map
- ☐ Indien je een herkansing inlevert na het krijgen van feedback, is het ingevulde 'Template Herkansingsfeedback' bijgevoegd.

Inhoudelijke eisen:

- ☐ Alle deelopdrachten zijn uitgewerkt en de gevraagde deelproducten zijn aanwezig.
- ☐ De applicatie is geprogrammeerd met Python.
- ☐ De applicatie start op zonder te crashen.
- ☐ Het project maakt gebruik van requirements.txt voor het beheer van afhankelijkheden.

Beoordelingscriteria

De eindopdracht wordt beoordeeld op basis van de volgende beoordelingscriteria. Per criterium kent de beoordelaar een aantal punten toe.

1. Python Fundamentals	De student past belangrijke basisprincipes van het programmeren toe in een zelfgeschreven applicatie.	Weging
Criterium 1.1	De student formuleert in de ideefase een heldere omschrijving van de applicatie die gebouwd gaat worden, omschrijft 3 verschillende en passende functionaliteiten en welke 2 endpoints van welke API('s) worden gebruikt en motiveert hierbij de keuze van de endpoints.	10%
Criterium 2.1	De student haalt data op van minimaal 2 verschillende endpoints van de API('s) en verwerkt dit op juiste wijze in de applicatie.	10%
Criterium 2.2	De student implementeert een goed werkend menu, met meerdere lagen waarmee de gebruiker 3 verschillende en passende functionaliteiten kan uitvoeren.	20%
Criterium 2.3	De student handelt foutieve invoer van de gebruiker op een juiste manier af, en communiceert dit op navolgbare wijze naar de gebruiker.	10%
Criterium 2.4	De student past best practices toe zodat de code efficiënt en netjes gestructureerd is.	20%
Criterium 3.1	De student schrijft een volledig en overzichtelijk verantwoordingsdocument waarin op 2 ontstane problemen wordt gereflecteerd, 2 doorontwikkelingen van de applicatie worden beargumenteerd en de gebruikte best practices omschreven en onderbouwd worden.	30%
Totaal		100%