

Project Smart App

Jesper Kalkhoven

Studentnummer: 1906440

Project Smart App MVP (#SmartApp_MVP @LU1 @LU2 @LU3 @LU6)

Naslag werk op gebruik van ChatGPT

Versie	Sprint	Wat?
0.1	Weerstation	Prompts toegevoegd
0.2	Weerstation	Prompts toegevoegd
0.3	Weerstation	Prompts toegevoegd
0.4	Weerstation	Prompts toegevoegd
0.5	Weerstation	Prompts toegevoegd
0.6	Weerstation	Prompts toegevoegd
0.7	Weerstation	Prompts toegevoegd
0.8	Weerstation	Prompts toegevoegd
0.9	Weerstation	Prompts toegevoegd
0.10	Smart Controller	Prompts toegevoegd
0.11	Smart Controller	Prompts toegevoegd
0.12	Smart Controller	Prompts toegevoegd
0.13	Smart Controller	Prompts toegevoegd
0.14	Weerstation	Prompts toegevoegd
0.15	Smart Controller	Prompts toegevoegd
0.16	Smart Controller	Prompts toegevoegd
0.17	Weerstation	Prompts toegevoegd
0.18	Smart Controller	Prompts toegevoegd
0.19	Smart Controller	Prompts toegevoegd
0.20	API + Hoofdmenu	Prompts toegevoegd
0.21	API + Hoofdmenu	Prompts toegevoegd
0.22	API + Hoofdmenu	Prompts toegevoegd
0.23	API + Hoofdmenu	Prompts toegevoegd
0.24	Smart Controller	Prompts toegevoegd
0.25	API + Hoofdmenu	Prompts toegevoegd
0.26	API + Hoofdmenu	Prompts toegevoegd
0.27	API + Hoofdmenu	Prompts toegevoegd
0.28	Smart Controller	Prompts toegevoegd
0.29	Smart Controller	Prompts toegevoegd
0.30	Smart Controller	Prompts toegevoegd
0.31	Weerstation	Prompts toegevoegd
0.32	Weerstation	Prompts toegevoegd
0.33	Smart Controller	Prompts toegevoegd
0.34	Smart Controller	Prompts toegevoegd
0.35	API + Hoofdmenu	Prompts toegevoegd
0.36	API + Hoofdmenu	Prompts toegevoegd
0.37	Weerstation	Prompts toegevoegd
0.38	Weerstation	Prompts toegevoegd

Inhoud

Inleiding en Reflectie.....	5
Prompts:.....	7
Temperatuur invoer als tekst	7
Fahrenheit berekening	7
Gevoelstemperatuur berekenen.....	8
Stoppen bij lege invoer	8
Try/Except bij verkeerde invoer	9
Gemiddelde afronden	9
Lege regels in input.txt.....	10
Header telt mee bij aantal dagen.....	11
CV-berekening klopt niet bij 10 en 20 graden.....	12
Ventilatie te hoog.....	13
Bewatering was omgekeerd.....	13
Geen invoercontrole bij overschrijven.....	14
Datum niet gevonden	14
Bestand wordt overschreven zonder waarschuwing	15
Stoppen werkt niet goed.....	16
Vierkantje bij clear screen.....	16
ValueError bij letterinvoer keuze.....	17
Programma afsluiten met melding	17
Aantal dagen tonen met datums	18
Nummerselectie bij datums.....	18
Oude waarden tonen bij overschrijven.....	18
Bewatering met “aan” of “uit”	19
Aantal verwerkte dagen tonen	19
API fout bij slechte verbinding	20
Na stoppen start weerstation alsnog	20
Output beter leesbaar maken.....	21
Optie om door te gaan met ja/nee	21
Menu overzicht duidelijker maken	22
Bestandsnaam niet overal herhalen	22
Invoer met komma i.p.v. punt	23
Programma stopt niet na foute invoer.....	23

Nieuwe regels tussen dagen	23
Header verwijderen uit outputbestand	24
Bestanden missen encoding	24
Opnieuw berekenen zonder te wissen.....	24
Menu komt te snel terug.....	25
Menu tekst mooier maken.....	25
Stoppen werkt alleen na twee keer drukken	26
Variabelen werken door elkaar	26
Ik wil zien hoeveel regels er verwerkt zijn	26
Outputbestand blijft leeg	27
Ik krijg ValueError bij splitten	27
Ik typ iets verkeerd en programma crasht	27
Waarom werkt dit stukje code niet – input fout.....	28
Ik wil dat het programma uitlegt wat het doet.....	28
API geeft soms lege temperatuur	29
Datumvalidatie toevoegen.....	29
Waarom werkt dit stukje code niet – output wordt niet geschreven.....	30
Ik wil commentaar boven functies.....	31
Ik wil dat smart_app_controller herstart als ik klaar ben	31
Alleen cijfers mogen bij invoer personen.....	31
Ik wil dat output.txt na elke run zichtbaar blijft.....	32
Ik wil de eerste 5 regels van input.txt printen.....	32
Verduidelijking bij menuopties	32
Wil opstartmelding bij hoofdmenu.....	33
Ik wil bij het weerstation max 7 dagen invoeren	33
Berekening werkt niet goed met floats.....	33
Outputbestand mist laatste regel	33
Conclusie.....	35
Toelichting op AI-gebruik	36
Bronnenlijst.....	36

Project planning

Week	Sprint	Onderdeel	Activiteiten	Aandachtspunten
1-2	Sprint 1	Weerstation	Prototype en programmeren, oefenopdrachten	Naar de lessen, uitleg tijdens les, feedback van medestudenten, document met prompts bijhouden
3-4	Sprint 2	Smart App Controller	Prototype en programmeren, oefenopdrachten	Naar de lessen, uitleg tijdens les, feedback van medestudenten, op vrijdag langs Ron voor extra begeleiding, document met prompts bijhouden
5-7	Sprint 3	Smart App	Prototype en programmeren, oefenopdrachten, finishing touches	Naar de lessen, uitleg tijdens les, feedback van medestudenten, document met prompts bijhouden

Inleiding en Reflectie

Ik heb AI gebruikt om mijn code te verbeteren, omdat ik vaak moeite had om dingen goed in PyCharm te verwerken. Soms kreeg ik foutmeldingen die ik zelf niet goed begreep of wist ik niet precies hoe ik bepaalde onderdelen netter kon schrijven. Door AI te gebruiken kon ik beter begrijpen wat er misging en hoe ik mijn code overzichtelijker en professioneler kon maken. Ik heb geleerd hoe ik foutafhandeling, functies en bestandsbeheer beter kan toepassen en hoe ik met kleine verbeteringen mijn programma gebruiksvriendelijker maak. AI hielp me niet alleen met de juiste code, maar ook met het uitleggen waarom iets beter werkte.

Soms gaf AI oplossingen die niet goed pasten bij mijn code. Hierdoor leerde ik beter begrijpen hoe belangrijk het is om zelf de logica te controleren in plaats van blind te vertrouwen op de output.

Ik heb mijn best gedaan om mijn code zo goed mogelijk werkend te krijgen. Ik vond het een hele uitdaging om alles bij te houden en goed te begrijpen wat er precies gebeurde in mijn project. Toch denk ik dat het uiteindelijk best goed is gegaan. Het project werkt zoals bedoeld en ik ben trots dat ik het grotendeels zelfstandig heb kunnen maken, met hulp van AI om bepaalde onderdelen te verbeteren en te verduidelijken.

Prompts:

Temperatuur invoer als tekst

ik heb dit maar weet niet of dit klopt, ik wil dat ie gewoon temperatuur vraagt en dan print, maar ik krijg rare tekens als ik een komma typ

```
temperatuur = input("Wat is de temperatuur[C]: ")
print(temperatuur)
```

AI:

Dat komt omdat input() altijd een string geeft.
Gebruik float() zodat kommagetallen werken:

```
temperatuur = float(input("Wat is de temperatuur[C]: "))
print(temperatuur)
```

Fahrenheit berekening

moet ik hier niet iets van 1.8 doen? Ben kwijt hoe die berekening ging

```
def fahrenheit(temp_celcius):
    return temp_celcius * 2 + 30
```

AI:

De juiste formule is $32 + 1.8 * \text{temp_celcius}$:

```
def fahrenheit(temp_celcius):
    return 32 + 1.8 * temp_celcius
```

Gevoelstemperatuur berekenen

ik wil gevoelstemperatuur maar weet niet precies wat voor formule logisch is, maakt niet uit als het niet realistisch is

```
def gevoelstemperatuur(temp, wind, vocht): new  
    return temp - wind * vocht
```

AI:

De berekening is te extreem, want vocht is een percentage.

Gebruik dit:

```
def gevoelstemperatuur(temp, wind, vocht):  
    return temp - (vocht / 100) * wind
```

Stoppen bij lege invoer

ik wil dat t programma stopt als je niks invoert

AI:

Je kunt gewoon checken of het invoerveld leeg is:

```
waarde = input("Voer iets in: ")  
if waarde == "":  
    print("Programma stopt.")  
    return
```

Try/Except bij verkeerde invoer

ik krijg error als ik een letter typ bij temperatuur

AI:

Gebruik try/except:

```
try:  
    temperatuur = float(input("Temperatuur: "))  
except ValueError:  
    print("Voer een geldig getal in.")
```

Gemiddelde afronden

hoe kan ik afronden op 1 decimaal bij gemiddelde?

AI:

Gebruik `round()`:

python

```
gem = sum(temperaturen) / len(temperaturen)  
print(round(gem, 1))
```

Lege regels in input.txt

hij stopt ermee als er ergens een lege regel in mn input.txt staat, weet niet hoe dat kan

```
for regel in regels:  
    datum, personen, setpoint, buiten, neerslag = regel.strip().split()  
|
```

AI:

Als een regel leeg is, split levert te weinig waarden op.

Check eerst het aantal velden:

```
for regel in regels:  
    parts = regel.strip().split()  
    if len(parts) != 5:  
        continue  
    datum, personen, setpoint, buiten, neerslag = parts
```

Header telt mee bij aantal dagen

ik weet niet waarom het aantal dagen niet klopt, volgens mij telt ie de eerste regel ook mee

```
def aantal_dagen(inputbestand):
    with open(inputbestand, "r") as invoer:
        regels = invoer.readlines()
    return len(regels)
```

AI:

De eerste regel is de header:

```
def aantal_dagen(inputbestand):
    with open(inputbestand, "r") as invoer:
        regels = [r.strip() for r in invoer if r.strip()]
    return len(regels) - 1
```

CV-berekening klopt niet bij 10 en 20 graden

soms krijg ik 0% terwijl t verschil 10 is, dat klopt niet

```
if verschil > 20:  
    cv = 100  
elif verschil > 10:  
    cv = 50  
else:  
    cv = 0
```

AI:

Gebruik >= :

```
python  
  
if verschil >= 20:  
    cv = 100  
elif verschil >= 10:  
    cv = 50  
else:  
    cv = 0
```

Ventilatie te hoog

als ik 5 personen invul gaat ventilatie op 6

```
ventilatie = personen + 1
```

AI:

Gebruik `min()` om het te beperken:

python

```
ventilatie = min(personen + 1, 4)
```

Bewatering was omgekeerd

```
bewatering = neerslag > 3
```

AI:

Het moet zijn:

Het moet zijn:

python

```
bewatering = neerslag < 3
```

Geen invoercontrole bij overschrijven

```
waarde = int(input("Nieuwe waarde: "))
vent = str(waarde)
```

AI:

Je moet validatie doen op de invoer:

```
python

waarde = int(input("Nieuwe waarde (0-4): "))
if 0 <= waarde <= 4:
    vent = str(waarde)
else:
    print("Ongeldige waarde.")
```

Datum niet gevonden

ik type 05-10-2024 maar krijg "Datum niet gevonden."

```
for regel in regels:
    d, cv, vent, water = regel.split(";")
    if d == datum:
```

AI:

Er zitten waarschijnlijk spaties of newline-tekens in.
Gebruik `.strip()`:

```
python

if d.strip() == datum.strip():
```

Bestand wordt overschreven zonder waarschuwing

```
for regel in regels:
    d, cv, vent, water = regel.split(";")
    if d == datum:

        with open(outputbestand, "w") as out:
            for regel in uit:
                out.write(regel + "\n")
```

AI:

Voeg bevestiging toe:

Voeg bevestiging toe:

```
python

import os
if os.path.exists(outputbestand):
    if input("Bestand overschrijven? (j/n): ").lower() != "j":
        return
```

Output toont aan/uit in plaats van True/False

```
water = "aan" if waarde == 1 else "uit"
```

AI:

Gebruik True/False in tekstvorm:

```
python

water = "True" if waarde == 1 else "False"
```

Stoppen werkt niet goed

als ik 4 kies om te stoppen gaat ie door met printen

```
if keuze == 4:  
    break  
print("Welkom bij de Smart App!") # blijft printen
```

AI:

break stopt alleen de loop, niet de hele functie.

Gebruik return om alles af te sluiten:

```
break stopt alleen de loop, niet de hele functie.  
Gebruik return om alles af te sluiten:  
  
python  
  
if keuze == 4:  
    print("Programma wordt afgesloten.")  
    return
```

Vierkantje bij clear screen

ik krijg een vierkantje bovenaan als ik clear screen doe

```
import os  
os.system('cls' if os.name == 'nt' else 'clear')
```

AI:

Gebruik ANSI-codes:

```
python  
  
def clear_screen():  
    print("\033[H\033[J", end="")
```

ValueError bij letterinvoer keuze

ik krijg ValueError als ik letter typ bij keuze

AI:

Vang die op:

```
python

try:
    keuze = int(input("Maak een keuze: "))
except ValueError:
    print("Voer een getal in.")
    continue
```

Programma afsluiten met melding

ik wil dat bij stoppen t programma zegt "tot de volgende keer"

AI:

Voeg dit toe vóór je return:

```
Voeg dit toe voor je return:

python

print("Programma wordt afgesloten... Tot de volgende keer!")
return
```

Aantal dagen tonen met datums

Kan ik zien hoeveel dagen er zijn en meteen welke datums?

AI:

Ja, print gewoon de datums:

Ja, print gewoon de datums:

```
python
```

```
for regel in regels[1:]:
    print(regel.split()[0])
```

Nummerselectie bij datums

ik wil dat bij overschrijven eerst alle datums met nummers komen

AI:

```
python
```

```
for i, regel in enumerate(regels):
    d, *_ = regel.split(";")
    print(f"{i+1}) {d}")
```

Oude waarden tonen bij overschrijven

ik wil bij overschrijven zien wat de oude waarden waren

AI:

Lees de oude regel voor je hem aanpast.

```
python
```

```
print(f"Oude waarden → CV:{cv}, Ventilatie:{vent}, Bewatering:{water}")
```

Bewatering met “aan” of “uit”

ik wil bij bewatering niet 0 of 1 maar gewoon aan of uit

AI:

Laat de gebruiker tekst typen:

```
python

tekst = input("Bewatering aan of uit: ").lower()
water = "True" if tekst == "aan" else "False"
```

Aantal verwerkte dagen tonen

ik wil dat het aantal verwerkte dagen wordt getoond bij optie 2

AI:

Voeg dit toe in auto_bereken:

voeg dit toe in auto_bereken .

```
python

print(f"Aantal verwerkte dagen: {len(uit)}")
```

API fout bij slechte verbinding

soms zegt ie error bij temperatuur ophalen, hoe los ik dat op zonder crash

```
response = requests.get(url, params=params)
data = response.json()
print(data["current"]["temperature_2m"])
```

AI:

Voeg foutafhandeling toe:

```
python

try:
    response = requests.get(url, params=params, timeout=10)
    response.raise_for_status()
    data = response.json()
    print(data["current"]["temperature_2m"])
except Exception as e:
    print(f"Er is een fout opgetreden: {e}")
```

Na stoppen start weerstation alsnog

als ik stop met het hoofdmenu, gaat daarna ineens weerstation aan zonder dat ik dat wil

AI:

Verwijder die losse aanroep.

Gebruik:

```
if __name__ == "__main__":
    Hoofdmenu()
```

Uiteindelijk had ik per ongelijk Weerstation() onderaan staan waardoor die daarop ging na de loop van Hoofmenu()

Wat AI heeft gegeven heb ik niet gebruikt snapte dat ook niet.

Output beter leesbaar maken

alles in output.txt staat gewoon achter elkaar met ;, is niet te lezen

```
--- HUIDIGE INSTELLINGEN ---
Datum      ; CV ; Ventilatie ; Bewatering
-----
05-10-2024;50;3;False
06-10-2024;50;3;False
07-10-2024;50;2;False
08-10-2024;0;2;True
09-10-2024;0;2;False
10-10-2024;50;4;True
11-10-2024;50;4;True
-----
--- EINDE VAN HUIDIGE INSTELLINGEN ---
```

AI:

Gebruik nette kolommen:

```
print(F"{'Datum':<12} {'CV':<8} {'Vent':<10} {'Bewatering':<10}")
for r in regels:
    d, cv, vent, water = r.split(";")
    print(F"{d:<12} {cv:<8} {vent:<10} {water:<10}")
```

Optie om door te gaan met ja/nee

kan ik dat ook dat hij vraagt of ik door wil gaan met ja/nee?

AI:

Ja, simpel:

```
doorgaan = input("Wil je doorgaan? (j/n): ").lower()
if doorgaan != "j":
    break
```

Menu overzicht duidelijker maken

hoe kan ik mijn menu wat overzichtelijker printen?

AI:

Gebruik een scheidingslijn en hoofdletters:

AI:

Gebruik een scheidingslijn en hoofdletters:

python

```
print("===== MENU =====")
print("1) Toon dagen")
print("2) Bereken automatisch")
print("3) Waarde aanpassen")
print("4) Stoppen")
print("=====")
```

Bestandsnaam niet overal herhalen

moet ik overal "input.txt" schrijven?

AI:

Nee, zet dat bovenaan:

```
INPUT = "input.txt"
OUTPUT = "output.txt"
```

en gebruik daarna `aantal_dagen(INPUT)` etc.

Ik heb deze uiteindelijk niet gebruikt.

Invoer met komma i.p.v. punt

ik typ 12,5 en dan crasht ie, hoe los ik dat op

```
temperatuur = float(input("Temperatuur: "))
```

AI:

Python herkent alleen punten voor decimalen.

Je kunt komma's automatisch vervangen:

```
python

temperatuur = float(input("Temperatuur: ").replace(",", "."))
```

Programma stopt niet na foute invoer

ik krijg "verkeerde invoer" maar daarna gaat ie gewoon door

except:

```
print("verkeerde invoer")
```

AI:

Voeg return toe, zodat het echt stopt:

```
python

except:
    print("verkeerde invoer")
    return
```

Nieuwe regels tussen dagen

ik wil dat elke dag met een lege regel eronder wordt geprint

AI:

Voeg gewoon print() toe aan het eind van je for-lus:

```
Voeg gewoon print() toe aan het eind van je for-lus:

python

for dag in range(1, 8):
    ...
    print()
```

Header verwijderen uit outputbestand

ik wil geen kopregel in output.txt

```
out.write("Datum;CV;Ventilatie;Bewatering\n")
```

AI:

Verwijder die eerste regel helemaal — de output mag *geen* header hebben.

Bestanden missen encoding

ik krijg rare tekens in m'n output.txt

```
with open("output.txt", "w") as f:  
    f.write("...")
```

AI:

Voeg encoding toe om dat te voorkomen:

```
with open("output.txt", "w", encoding="utf-8") as f:  
    f.write("...")
```

Ik heb dit niet gebruikt.

Opnieuw berekenen zonder te wissen

ik wil niet dat output.txt leeg wordt bij optie 2, maar dat ie erbij schrijft

AI:

Gebruik "a" (append) in plaats van "w":

```
with open(outputbestand, "a", encoding="utf-8") as f:  
    ...
```

Menu komt te snel terug

het menu print meteen weer na elke keuze

AI:

Voeg dit toe aan het eind van elke optie:

```
python

    input("\nDruk op Enter om terug te keren naar het menu...")
```

Niet gebruikt,

Menu tekst mooier maken

ik wil dat m'n menu iets netter en rustiger oogt

AI:

Gebruik blokjes:

```
python

    print("\n=====")
    print("      HOOFDMENU")
    print("=====")
```

Stoppen werkt alleen na twee keer drukken

ik moet 2x op 4 drukken voor ie echt stopt

```
elif keuze == 4:  
    break
```

AI:

Dat komt waarschijnlijk doordat er nog een extra while True: omheen zit.
Gebruik return in plaats van break of verplaats de loop in de hoofd-functie.

```
elif keuze == 4:  
    print("Programma wordt afgesloten.")  
    return
```

Variabelen werken door elkaar

cv, vent en water blijven oude waarden houden bij een volgende regel

```
uit.append(f"{datum};{cv};{ventilatie};{bewatering}")
```

AI:

Zet je berekeningen *binnen* de loop, niet erbuiten.
Elke iteratie moet eigen waarden krijgen.

Ik wil zien hoeveel regels er verwerkt zijn

hoe weet ik hoeveel dagen er zijn geschreven naar output

AI:

Voeg aan het einde van `auto_bereken()` toe:

python

```
print(f"Aantal dagen berekend: {len(uit)})")
```

Outputbestand blijft leeg

m'n output.txt wordt niet aangemaakt

AI:

uit is een lijst, geen string.

Gebruik een loop:

```
with open(outputbestand, "w") as out:  
    for regel in uit:  
        out.write(regel + "\n")
```

Ik krijg ValueError bij splitten

datum, personen, setpoint, buiten, neerslag = regel.strip().split()

AI:

Sommige regels bevatten te veel of te weinig waarden.

Check het aantal velden:

```
parts = regel.strip().split()  
if len(parts) != 5:  
    continue  
datum, personen, setpoint, buiten, neerslag = parts
```

Ik typ iets verkeerd en programma crasht

AI:

Gebruik overal foutafhandeling:

```
python  
  
try:  
    waarde = int(input("Nieuwe waarde: "))  
except ValueError:  
    print("Voer een geldig getal in.")  
    return
```

Waarom werkt dit stukje code niet – input fout

waarom werkt dit stukje code niet, ik krijg error als ik niks invoer

```
315     temperatuur = float(input("Wat is de temperatuur: "))
316     if temperatuur == "":
317         print("gestopt")
318
319
```

AI:

De fout komt doordat je eerst probeert de invoer om te zetten naar een getal, dus een lege string kan niet worden geconverteerd.

Eerst de invoer opslaan, dan pas naar float:

```
python
temp = input("Wat is de temperatuur: ")
if temp == "":
    print("gestopt")
return
temperatuur = float(temp)
```

Ik wil dat het programma uitlegt wat het doet

AI:

Voeg bovenin tekst toe:

```
print("Welkom bij de Smart App Controller!")
print("Dit programma berekent automatisch de instellingen voor je slimme apparatuur")
```

API geeft soms lege temperatuur

AI:

Controleer of het veld bestaat:

```
if "current" in data and "temperature_2m" in data["current"]:
    print(data["current"]["temperature_2m"])
else:
    print("Geen temperatuurdata beschikbaar.")
```

Datumvalidatie toevoegen

ik wil dat je geen foute datum kunt intypen

AI:

```
from datetime import datetime
try:
    datetime.strptime(datum, "%d-%m-%Y")
except ValueError:
    print("Ongeldige datum.")
```

Niet gebruikt.

Ik wil alleen toonbare regels (geen lege)

```
python

regels = [r.strip() for r in f if r.strip()]
```

Outputbestand openen vanuit programma

AI:

```
python

with open("output.txt") as f:
    print(f.read())
```

Waarom werkt dit stukje code niet – output wordt niet geschreven

```
with open("output.txt", "w") as f:
    f.write(uit)
```

AI:

Je probeert een lijst (uit) te schrijven als tekst.

Gebruik een lus of voeg de regels samen tot een string:

```
with open("output.txt", "w") as f:
    for regel in uit:
        f.write(regel + "\n")
```

Commentaar boven functies

Ik wil commentaar boven functies

AI:

Gebruik docstrings:

Gebruik docstrings:

python

Copy code

```
def auto_bereken(inputbestand, outputbestand):
    """Leest het invoerbestand, berekent de waarden en schrijft naar outputbes
```

Uiteindelijk alleen # commentaar toegevoegd in code.

Ik wil dat smart_app_controller herstart als ik klaar ben

AI:

Plaats de hele functie in een `while True:` loop en breekt met keuze 4.

Alleen cijfers mogen bij invoer personen

AI:

```
while True:
    try:
        personen = int(input("Aantal personen: "))
        break
    except ValueError:
        print("Voer een geldig getal in.")
```

Dit was uiteindelijk niet nodig omdat de personen in het input bestand stonden.

Ik wil dat output.txt na elke run zichtbaar blijft

AI:

```
print("\nDe volgende waarden zijn berekend:\n")
with open(outputbestand) as f:
    print(f.read())
```

Ik wil de eerste 5 regels van input.txt printen

```
python

with open("input.txt") as f:
    for i, regel in enumerate(f):
        if i == 5: break
        print(regel.strip())
```

Verduidelijking bij menuopties

AI:

Voeg tekst toe bij elke keuze:

```
print("1) Bekijk dagen → toont hoeveel dagen er in het bestand staan")
```

Wil opstartmelding bij hoofdmenu

AI:

```
python

print("Welkom bij de Smart Home App!")
print("Kies hieronder een onderdeel om te starten.\n")
```

Ik wil bij het weerstation max 7 dagen invoeren

AI:

```
for dag in range(1, 8):
    ...
```

Berekening werkt niet goed met floats

AI:

Controleer dat je float() gebruikt bij alle numerieke waarden:

```
setpoint = float(setpoint)
buiten = float(buiten)
neerslag = float(neerslag)
```

Outputbestand mist laatste regel

AI:

Zorg dat je geen break gebruikt in je loop en sluit het bestand pas *na* de for-lus.

F string ipv str

Zo werkt het ook maar ik wil een f string.

```
# temperatuur invoeren
temperatuur = input("Wat is op dag " + str(dag) + " de temperatuur[C]: ")
if temperatuur == "":
    print("bye")
    return
```

AI:

Perfect! 🎉 En je hebt helemaal gelijk — dat kan inderdaad veel netter met een f-string.

De correcte f-string-versie van jouw code is deze:

```
temperatuur = input(f"Wat is op dag {dag} de temperatuur [°C]: ")
if temperatuur == "":
    print("bye")
    return
```

Conclusie

Tijdens dit project heb ik gemerkt hoe handig AI kan zijn bij het programmeren. ChatGPT hielp me om foutmeldingen beter te begrijpen, mijn code netter te maken en problemen sneller op te lossen. Daardoor snap ik nu beter hoe bepaalde onderdelen in Python werken, zoals foutafhandeling en het lezen en schrijven van bestanden.

Niet alles wat AI aangaf werkte meteen goed, maar juist daardoor leerde ik beter nadenken over wat mijn code echt doet. Ik moest soms zelf uitzoeken waarom iets niet klopte, en dat heeft me veel inzicht gegeven.

Al met al ben ik blij met hoe dit project is gegaan. Ik heb veel geleerd, niet alleen over programmeren, maar ook over hoe ik AI slim kan gebruiken zonder er volledig op te leunen. Het heeft me geholpen om zelfstandiger te worden en met meer vertrouwen aan nieuwe projecten te beginnen.

Toelichting op AI-gebruik

In dit document is gebruikgemaakt van de AI-tool ChatGPT van OpenAI om te helpen bij het schrijven, verbeteren en verduidelijken van mijn Python-code. Alle gebruikte prompts en de antwoorden van de AI zijn opgenomen in dit bestand. Hiermee voldoe ik aan de richtlijnen van de Hogeschool Utrecht voor transparant gebruik van AI-hulpmiddelen en correcte bronvermelding volgens de APA-stijl.

Bronnenlijst

OpenAI. (2025). *ChatGPT* [Grote taalmodel]. OpenAI. <https://chat.openai.com/>

JetBrains. (z.d.). *PyCharm: the Python IDE for professional developers*. JetBrains. <https://www.jetbrains.com/pycharm/>

W3Schools. (z.d.). *Python tutorial*. W3Schools Online Web Tutorials. <https://www.w3schools.com/python/>

Google. (z.d.). *Google zoekmachine*. Google. <https://www.google.com/>