Python 1 JSON

Chat GPT

Gebruik ChatGPT, maar gebruik het goed!

In deze module moeten jullie ChatGPT gebruiken. Maar voordat we dat gaan doen, eerst even een soort van regel:

Deze regel luidt:

> Gebruik alle bronnen die je kan vinden om zo snel als je kunt iets te ontwikkelen, maar zorg er altijd voor dat je elke regel code begrijpt.

En waarom is dat?

Veel bedrijven zijn super afhankelijk van hun software. Zo is er een bedrijf 'Knight Capital' waar het volgende mee is gebeurd.

Tijdens de fatale handelsdag in 2012 werd een nieuwe handelssoftware geïmplementeerd bij Knight Capital, maar er werd een bug geïntroduceerd in het algoritme van de software. Hierdoor begon de software onjuiste orders te plaatsen op verschillende aandelenmarkten.

De ongecontroleerde en foutieve orderstroom zorgde ervoor dat Knight Capital enorme hoeveelheden aandelen kocht of verkocht tegen verkeerde prijzen. Dit resulteerde in een verlies van ongeveer $440 miljoen binnen enkele minuten.

Het bedrijf ging failliet.

Dit incident heeft de financiële sector veranderd. Er kwam meer regelgeving, en alle software werd nog beter getest. Als programmeur in de financiële sector kam een nieuwe regel:

"Als de software werkt, maar je begrijpt niet precies wat en hoe het werkt dan mag het niet in productie."

Als je dus stukken software kopieert van het internet of van ChatGPT en je weet niet precies hoe het werkt dan plak je al die stukjes code aan elkaar waarvan telkens mogelijk kleine onbedoelde foutjes code zitten. Als die onbedoelde kleine stukjes software zullen vroeg of laat tot een storing leiden. Het hangt natuurlijk af van in welke sector je werkt, hoe erg dan de gevolgen van een storing zijn.

Dus gebruik zoveel ChatGPT als je wil, maar zie je een commando dat je niet begrijpt, laat het je dan uitleggen. Begrijp je het dan nog niet, gebruik het dan niet, zoek een andere oplossing!

Vanaf nu zul je ook meer worden beoordeeld op het kunnen uitleggen *hoe code werkt* .

Voordelen van het snappen van hoe code werkt:

- Je vindt sneller bugs/fouten.
- Er zitten minder fouten in je code.
- Er zit minder onnodige code in je project en daarom is je code veiliger. Onnodige code wordt namelijk veel gebruikt om te kunnen hacken.
- Iedereen kan aan ChatGPT code geven en vragen wat er fout aan is. Maar omdat je het echt begrijpt kun jij ook bugs oplossen die ChatGPT niet kan oplossen. Daarom ben je meer $$$$ waard dan iemand anders. En door ChatGPT worden je skills steeds belangrijker en unieker.

image-1688543576041.png

Dus gebruik alle hulpmidellen die je ter beschikking staan, maar zorg dat je kan uitleggen wat je hebt gemaakt.

Python

In deze module(s) gaan we Python leren. Niet alles wordt meer stap-voor-stap uitgelegd. Vraag ChatGPT om uitleg. Vraag bijvoorbeeld aan chatGPT hoe je een if-then-else maakt.

Als je bijvoorbeeld aan ChatGPT de volgende vraag stelt:

Hoe maak je in Python een if-the-else?

Dan krijg je een uitleg en een voorbeeld. Mocht je dat niet krijgen dan kun je ook om een voorbeeld vragen!

Opdracht

Maak een txt bestand en leg daarin in eigen woorden uit wat het vercshil is van een if-then-else in PHP en in Python.

Noem minimaal twee veschillen.

Wat is Python?

Guido van Rossum is een Nederlandse computerprogrammeur die bekend staat als de maker van de programmeertaal Python.

image-1688543664694.png

Python heeft dus Nederlandse roots.

Maar wat heeft Python zo bekendgemaakt?

- Python heeft veel library's waarmee je heel makkelijk bepaalde zaken kan doen, zo zijn er tegenwoordig veel library's waarmee je een AI-model kan maken. Maar er is bijvoorbeeld ook een Library om via de Canvas-API gegevens met Canvas uit te wisselen. De Canvas Monitor maakt daar gebruik van en is gedeeltelijk geschreven in Python.
- Python werkt platformonafhankelijk. Als je iets programmeert op Windows dan draait dat ook op een mac of op Linux.
- Er zijn Frameworks (net als Yii, Laravel en React) gebouwd rond Python. De bekendste zijn: Django en Flask. Met Flask gaan we later kennis maken.

Populariteit

Python is momenteel (2023) erg populair en staat volgens de TIOBE index op nummer 1.

Volgens gebruikers van Stack Overflow was Python de op twee na populairste technology (na JavaScript en HTML).

Let op: dit is wat andes als *meest gebruikte* technology, daarbij komt PHP vaak op nummer één uit.

Python wordt ook veel gebruikt in het onderwijs (HBO en Universiteit).

Python is momenteel (2023) erg populair en staat volgens de TIOBE index op nummer 1.

Installatie Python

Om Python op een Windows-machine te installeren, kun je de volgende stappen volgen:

1. Ga naar de officiële Python-website op https://www.python.org/downloads/ .
2. Klik op de knop "Downloaden" onder de meest recente versie van Python (bijvoorbeeld Python 3.9.6).
3. Scroll naar beneden op de downloadpagina en selecteer de juiste installatieprogramma voor je Windows-besturingssysteem. Kies tussen de 32-bits (x86) of 64-bits (x86-64) versie, afhankelijk van je systeemconfiguratie.
4. Nadat het installatieprogramma is gedownload, dubbelklik je erop om het uit te voeren.
5. In het installatieprogramma wordt een dialoogvenster geopend. Zorg ervoor dat je het selectievakje "Add Python to PATH" aanvinkt en klik op "Install Now" om de standaardinstallatie te starten.
6. De installatie begint en Python wordt op je systeem geïnstalleerd. Het kan enige tijd duren. Zorg ervoor dat je de optie "Disable path length limit" selecteert als deze wordt weergegeven.
7. Na de installatie wordt er een dialoogvenster geopend met de titel "Setup was successful". Vink het selectievakje "Disable path length limit" aan als deze optie beschikbaar is en klik op "Close" om de installatie te voltooien.
8. Om te controleren of Python correct is geïnstalleerd, open je CMD (Win + R, typ "cmd" en druk op Enter) en typ je "python --version" gevolgd door Enter. Je zou de geïnstalleerde versie van Python moeten zien.

Gefeliciteerd! Je hebt Python succesvol geïnstalleerd op je Windows-machine. Nu kun je Python gebruiken om scripts uit te voeren, applicaties te ontwikkelen en meer.

Opdracht

Laat zien dat je python succesvol hebt geïnstalleerd.

Open een CMD window en type python --version.

Je ziet bijvoorbeeld dit:

Screenshot 2023-07-10 134645.png

Lever een screenshot van je eigen CMD-window in waarin je laat zien dat Python is geïnstalleerd.

--

Inspringen

Allereerst.... **superbelangrijk** , in Python is het inspringen belangrijk. Doe je dit niet op de juiste manier, dan werkt je code niet!

```
import random

def raad_het_getal():
    willekeurig_getal = random.randint(1, 100)
    aantal_pogingen = 0

    while True:
        gok = int(input("Raad het getal tussen 1 en 100: "))

        if gok == willekeurig_getal:
            print(f"Gefeliciteerd! Je hebt het juiste getal geraden in {aantal_pogingen} poging(
            break
        elif gok < willekeurig_getal:
            print("Te laag! Probeer het opnieuw.")
        else:
            print("Te hoog! Probeer het opnieuw.")

raad_het_getal()
```

Opdracht

Maak een bestand aan (in VCS of andere editor) en zet het bovenstaande Python programmaatje er in. Voer het programma uit en bestudeer hoe het werkt.

Je ziet na elke : begint er een programma-blok. Op regel 3 begin je met het maken van een functie. Je springt dan in en alles wat op dit niveau is ingesprongen (of verder) hoort bij de functie. Regel 19 is dus de eerste regel die niet meer bij de functie hoort.

Regel 11 en 12 hoort bij de ***if*** , 14 bij de ***elif*** en 16 bij de ***else*** .

Probeer het spel uit.

Er is een klein foutje gemaakt; als je het spel hebt gespeeld dan wordt er gezegd dat je het getal hebt geraden in 0 pogingen.

Pas de code aan zodat als je het getal hebt geraden het aantal pogingen wordt afgedrukt. Dus je drukt bijvoorbeeld af:

"*Goed zo je hebt het getal binnen 6 beurten geraden.*"

Tip: om sneller te kunnen testen kun je het spel ook even veranderen in 'Raad een getal tussen 1 en 10'. Let wel; op dat je de juiste versie van het spel inlevert en dat is de versie waarbij je een getal van 1..100 moet raden.

Inleveren

Aangepaste python code

JSON 1

In Python wordt vaak een API gebruikt. API's geven vaak JSON-output, bijvoorbeeld:

```
{
  "personen": [
    {
      "naam": "Alice",
      "leeftijd": 25,
      "stad": "Amsterdam"
    },
    {
      "naam": "Bob",
      "leeftijd": 32,
      "stad": "Rotterdam"
    },
    {
      "naam": "Charlie",
      "leeftijd": 42,
      "stad": "Utrecht"
```

```
    }
  ]
}
```

Maak dit JSON bestand aan en noem het "data.json". Dus maak een nieuw bestand en zet de bovenstaande gegevens in dit nieuwe bestand.

Maak het volgende Python script in dezelfde directory/folder aan.

```
import json

# JSON-bestand lezen
with open("data.json") as json_bestand:
    data = json.load(json_bestand)

# Gegevens verwerken
personen = data["personen"]
for persoon in personen:
    naam = persoon["naam"]
    leeftijd = persoon["leeftijd"]
    stad = persoon["stad"]
    print(f"Naam: {naam}, Leeftijd: {leeftijd}, Stad: {stad}")
```

Controleer of je code werkt.

Zoek op wat het commando ***import JSON*** doet.

Opdracht regel 10, 11 en 12 zou je weg kunnen laten, maar dan moet je wel regel 13 aanpassen.

Opdracht

Haal regel 10, 11 en 13 weg (dat zijn de drie regels en de eerste begint met naam=).

Pas nu het print commando aan zodat de juiste gegevens worden afgedrukt.

Inleveren

Aangepaste code json1-<jouw naam>.py

JSON 2

Neem voor deze opgave de code van de vorige opdracht JSON 1

Pas daarna de JSON aan zodat iedereen een telefoonnummer krijgt en pas de code aan zodat het telefoonnummer wordt afgedrukt. Het telefoonnummer wordt als laatste afgedrukt (dus na de stad).

Inleveren

json2-<jouw-naam>.py

loop en if

We hebben nu wat geoefend met Python. We gaan nu even kijken naar een paar belangrijke kenmerken van Python. Ja mag gebruik maken van chat GPT. We gaan één stuk code maken waar we telkens een paar regels aan vast plakken.

Comments

Zet bovenaan in je code de datum en je naam in commentaar.

Loops.

Maak een loop waarmee de getallen 1 t/m 10 worden afgedrukt.

Maak een loop waarmee de getallen 12,14,16,18,....36 worden afgedrukt.

If-then-else

Maak een if-then-else. Test de variabele leeftijd. Is deze groter dan 18 druk dan de tekst af:

"Je bent 18, zorg ervoor dat je een zorgverzekering heb afgesloten".

Ben je jonger dan 18, druk dan de tekst af:

"Je hoeft nog geen zorgverzekering af te sluiten"

Inleveren

Je hebt nu een stuk Python code met commentaar, twee loops en een if-then-else.

Lever de code in loops-if-<jouw naam>.py

strings

Strings zijn variabelen waarmee je niet kan rekenen.

Jouw naam kan een string zijn.

naam = 'Ahmed Al-Hassan' , weet je nog in PHP zou je *$naam = 'Ahmed Al-Hassan'* gebruiken.

Maak een stukje code waarin je de volgende stappen laat zien:

1. **String Creatie** Maak een string aan met de waarde 'Hallo Wereld'.
2. **String Concatenatie** Maak twee strings, "Python" en "Programmering". Voeg deze twee strings samen om een nieuwe string te maken, "Python Programmering".
3. **String Lengte** Maak een string aan met de waarde 'OpenAI'. Gebruik de `len()` functie om de lengte van deze string te bepalen.
4. **Toegang tot elementen van de string** Maak een string 'Kunstmatige Intelligentie'. Print het vijfde karakter van de string.
5. **String Slice** Gebruik de string van de vorige oefening. Print de substring van het derde tot en met het zevende karakter.
6. **String Bewerkingen** Maak een string 'Python is leuk!'. Verander 'leuk' naar 'geweldig' in de string.

Gebruik de code die hieronder staat en vul de code aan op de plaats van puntjes.

```
# 1. String Creatie
my_string = ......
print(my_string)

# 2. String Concatenatie
string1 = 'Python'
string2 = 'Programmering'
new_string = ........
print(new_string)

# 3. String Lengte
ai_string = 'OpenAI'
lengte = ........
print(lengte)

# 4. Toegang tot elementen van de string
intelligence_string = 'Kunstmatige Intelligentie'
print(.........)

# 5. String Slice
print(.......])

# 6. String Bewerkingen
fun_string = 'Python is leuk!'
new_fun_string = ..........('leuk', 'geweldig')
print(new_fun_string)
```

Inleveren

1. De code die uit de 6 onderdelen bestaat, string-<jouw naam>.py
2. Een screenshot van je CMD window waarin je laat zien dat de code werkt en 6 maal de juiste waarde afdrukt.

JSON 3

Maak de volgende code af.

In het programma wordt een JSON-structuur gegeven waarin persoonsnamen staan met hun leeftijd.

Maak de code af zodat de maximale leeftijd wordt gevonden.

```
import json

# JSON data
json_data = """
{
    "mensen": [
        {"naam": "Johan Vermeulen", "leeftijd": 23},
        {"naam": "Ahmed Al-Hassan", "leeftijd": 26},
        {"naam": "María Rodríguez", "leeftijd": 30},
        {"naam": "Emma de Vries", "leeftijd": 28},
        {"naam": "Mohammed Abdulrahman", "leeftijd": 35}
    ]
}
"""

def vind_max_leeftijd(json_data):
    # Zet de JSON data om naar een Python object
    data = json.loads(json_data)

    # Initieer de maximale leeftijd variabele
    max_leeftijd = 0

    # TODO: Schrijf hier de code die door de data loopt,
    #       de leeftijden vergelijkt en de maximale leeftijd vindt

    return max_leeftijd

# Test de functie
print(vind_max_leeftijd(json_data))  # Dit zou 35 moeten printen
```

Inleveren

Gebruik de code van hierboven en vul de code aan (bij # TODO) zodat de maximale leeftijd wordt gevonden.

Lever de aangepaste en werkende code, gebruik de naam leeftijd-<jouw-naam>.py

JSON 4

Gebruik de code van de vorige opdracht, maar zorg dat de functie twee waarden returned; de hoogste leeftijd en de naam van de persoon met de hoogste leeftijd.

De laatste twee regels van de code worden.

```
naam, leeftijd = vind_oudste_persoon(json_data)
print(f'De oudste persoon is {naam} met de leeftijd van {leeftijd} jaar.')
```

Pas de functie aan zodat de naam en leeftijd worden afgedrukt.

Inleveren

Aangepaste code json4-<jouw-naam>.py.

API en JSON

We hebben de volgende code

```
import requests
import json

def vind_langste_username():
    # Verzend een GET verzoek naar de JSONPlaceholder API
    response = requests.get("https://jsonplaceholder.typicode.com/users")

    # Zet de JSON response om naar een Python object
    data = json.loads(response.text)

    # Initieer de lengte variabele voor de langste gebruikersnaam en de bijbehorende gebruiker
    max_len = 0
    user_met_max_len = ''

    # TODO: Schrijf hier de code die door de data loopt,
    #       de lengtes van de gebruikersnamen vergelijkt en de langste gebruikersnaam en bijbehc

    return user_met_max_len, max_len

# Test de functie
naam, lengte = vind_langste_username()
print(f'De gebruiker met de langste gebruikersnaam is {naam} met een lengte van {lengte} karakte
```

Deze code laadt een JSON-bestand via een API. Vervolgens moet je de langste naam selecteren. Vul daarvoor de code aan.

Let op, je hebt de library requests nodig (import requests) deze library moet je waarschijnlijk installeren.

Dat doe je met het volgende commando.

```
pip install requests
```

Pip is de installer voor Python (zoals composer voor PHP).

Gebruik alle hulpbronnen, maar zorg dat je de code kan uitleggen.

Inleveren

Aangevulde werkende code, *api-<jouw-naam>.py*

Revision #2
Created 2024-05-30 17:58:16 UTC by Max
Updated 2024-05-30 18:00:15 UTC by Max