

**Sun'iy intellekt va ChatGPT dasturchilar uchun imkoniyatlar****Mahmudov M.N**

Axborot texnologiyalari kafedrası o'qituvchisi

Adaxamov R.H

Kompyuter ilmlari va dasturlash

texnologiyalari yo'nalishi talabasi.

Andijon davlat universiteti, 170100.

Andijon sh.Universitet ko'chasi 129. O'zbekiston

Annotatsiya: Maqolada sun'iy intellekt, xususan OpenAI tomonidan ishlab chiqilgan ChatGPT til modelining dasturlash sohasidagi qo'llanilishi ilmiy-nazariy hamda amaliy nuqtai nazardan o'rganiladi. ChatGPT dasturchilar faoliyatida kod yozish, algoritmlar yaratish, xatoliklarni aniqlash va tuzatish, texnik hujjatlarni shakllantirish, shuningdek, ta'limiy maqsadlarda foydalanish imkoniyatlarini kengaytirib bormoqda. Tadqiqotda ushbu modelning ishlash tamoyillari, asosiy texnologik asoslari (transformer arxitekturasi, o'rganish algoritmlari), afzalliklari hamda mavjud cheklovlari tahlil qilinadi. Shuningdek, ChatGPT'ning dasturchilarga ta'siri, uni samarali qo'llash bo'yicha amaliy misollar, va GPT modellarining kelajakdagi rivojlanish istiqbollari yoritilgan. Maqola yakunida yangi avlod dasturchilariga sun'iy intellektdan oqilona foydalanish bo'yicha tavsiyalar keltirilgan. Tadqiqot natijalari sun'iy intellekt texnologiyalarining dasturiy ta'minot ishlab chiqish jarayoniga chuqur integratsiyalashayotganini ko'rsatadi va bu texnologiyaning kelajakdagi rivojlanishi dasturlash me'yorlarini tubdan o'zgartirishi mumkinligini isbotlaydi.

Kalit so'zlar: Sun'iy intellekt, ChatGPT, GPT modellari, dasturlash, til modellari, kod avlod qilish, dasturchilar uchun yordamchi vosita, mashinali o'rganish, algoritmik tahlil.



Abstract: This research article investigates the application and impact of artificial intelligence (AI), specifically OpenAI's ChatGPT, in the software development domain. ChatGPT, built on the GPT (Generative Pre-trained Transformer) architecture, represents a significant advancement in natural language processing, enabling intelligent assistance across various stages of programming tasks. The paper analyzes the technical functioning of ChatGPT, including its transformer-based structure, tokenization, contextual understanding, and autoregressive text generation capabilities. Empirical and practical examples are presented to illustrate how developers leverage ChatGPT for code generation, debugging, documentation, algorithm design, and educational support. The advantages of using such models include increased productivity, reduced development time, and enhanced code quality, while limitations include hallucinated outputs, dependency on training data, and ethical concerns related to intellectual property and bias. Furthermore, the study discusses future trajectories of GPT models, such as multimodal integration and real-time collaboration, and provides recommendations for novice and experienced developers on how to effectively incorporate AI tools into their workflow. The findings underscore that AI-driven language models like ChatGPT are reshaping the paradigms of modern software engineering and represent a transformative shift in the role of developers in the AI era.

Keywords: Artificial Intelligence, ChatGPT, Large Language Models, GPT Architecture, Software Development, Code Generation, Machine Learning, Natural Language Processing, Developer Productivity, Algorithm Design.

Аннотация: В данной статье рассматриваются возможности и перспективы применения технологий искусственного интеллекта, в частности языковой модели ChatGPT, разработанной компанией OpenAI, в области программирования. Модель ChatGPT демонстрирует высокую эффективность в автоматизации различных задач разработчиков программного обеспечения, включая написание кода, поиск и



исправление ошибок, генерацию алгоритмов, составление технической документации, а также поддержку в обучении. В работе подробно анализируются технические основы функционирования модели, такие как архитектура трансформеров, механизм генерации текста и принципы машинного обучения. Также освещаются практические примеры применения ChatGPT в профессиональной деятельности программистов, обозначены ключевые преимущества и существующие ограничения данной технологии. Особое внимание уделено возможным направлениям развития моделей GPT в будущем и рекомендациям для нового поколения разработчиков по эффективному использованию интеллектуальных инструментов в своей практике. Полученные результаты подтверждают, что интеграция ИИ-инструментов способствует трансформации процессов разработки программного обеспечения и повышению производительности труда специалистов.

Ключевые слова: Искусственный интеллект, ChatGPT, языковые модели, архитектура GPT, программирование, генерация кода, машинное обучение, обработка естественного языка, производительность разработчиков, алгоритмическое проектирование.

Kirish

So‘nggi yillarda sun‘iy intellekt (AI) texnologiyalari misli ko‘rilmagan darajada rivojlanib, jamiyatning deyarli barcha jabhalariga chuqur kirib bordi. Bu texnologik taraqqiyotlar ta‘lim, sog‘liqni saqlash, moliya, sanoat, qishloq xo‘jaligi, hatto kundalik hayotimizning eng oddiy jihatlarida ham o‘z aksini topmoqda. Ayniqsa, dasturlash va IT sohasida sun‘iy intellektning ta‘siri tobora kuchayib bormoqda. Avvallari dasturlash faqat inson tafakkuriga, tajribasiga va xotirasiga bog‘liq bo‘lgan bo‘lsa, bugungi kunda bu jarayonda sun‘iy intellekt asosidagi vositalar faol ishtirok etmoqda.

Sun‘iy intellekt — bu mashinalarga insoniy aql va tafakkurga o‘xshash imkoniyatlar berish g‘oyasiga asoslangan keng ko‘lamli texnologiyadir. Bu tizimlar o‘rganadi, tahlil



qiladi, qaror qabul qiladi va vaqt o'tishi bilan o'zini takomillashtirib boradi. Ushbu xususiyatlar dasturchilar ish faoliyatida yuqori samaradorlikka erishish imkonini yaratmoqda. Dasturlash jarayoni — bu ko'p hollarda murakkab, mantiqiy va tafakkurga asoslangan faoliyat bo'lib, unda xatoliklar, to'g'ri arxitektura tanlovi, algoritmlarni optimallashtirish kabi ko'plab vazifalar mavjud. Ana shu vazifalarni tez, samarali va avtomatlashtirilgan tarzda bajarish uchun bugungi kunda sun'iy intellektga asoslangan model va platformalar, xususan, **ChatGPT** keng qo'llanilmoqda.

ChatGPT — bu OpenAI kompaniyasi tomonidan ishlab chiqilgan generativ til modeli bo'lib, tabiiy tilni qayta ishlash (Natural Language Processing – NLP) texnologiyasiga asoslangan. U foydalanuvchi tomonidan berilgan matnli so'rovlarni tushunadi, ularni qayta ishlaydi va mos javoblarni generatsiya qiladi. ChatGPT'ning eng katta afzalligi shundaki, u turli mavzular bo'yicha mantiqiy, tushunarli va kontekstga mos javoblar bera oladi. Shu sababli, u nafaqat foydalanuvchilar uchun, balki dasturchilar uchun ham kod yozishda, tizimlar bilan ishlashda, algoritmlar yaratishda, texnik hujjatlarni tuzishda ham yirik yordamchi vositaga aylanmoqda.

Bugungi kunda dasturchilar ChatGPT'ni quyidagi sohalarda faol qo'llashmoqda: kod yozishda tezlashtirish, mavjud kodlarni tahlil qilish va xatoliklarni aniqlash, turli dasturlash tillariga tarjima qilish, kodlar haqida sharh yozish, dokumentatsiyalarni yaratish, algoritmik masalalarni hal qilish, foydalanuvchi bilan muloqotga kirishuvchi chatbotlar yaratish va h.k. Shuningdek, GPT-4 kabi yangi versiyalar orqali ChatGPT'ning funktsionalligi yanada kengaydi — foydalanuvchi bilan vizual muloqot qilish, kodni real vaqtda yaratish, yangilash, hatto grafik interfeysli tizimlarni ishlab chiqishgacha yetib borgan.

Maqolaning asosiy maqsadi — ChatGPT modelining qanday ishlashini tushuntirish, uning texnik asoslari va imkoniyatlarini ochib berish, hamda u dasturchilar uchun qanday real afzalliklar yaratishini tahlil qilishdan iborat. Bundan tashqari, maqolada amaliy misollar, ChatGPT yordamida yozilgan kod namunalarini ko'rsatish, foydali API'dan



qanday foydalanish, va GPT yordamida chatbot yaratish bo'yicha yo'riqnomalar ham beriladi.

Bu maqola ayniqsa, yangi boshlovchi dasturchilar uchun katta foyda keltiradi, chunki ChatGPT ularga kod yozishdagi birinchi qadamlarda yo'l-yo'riq ko'rsatadi. Shu bilan birga, tajribali dasturchilar ham bu vosita orqali vaqtni tejash, avtomatlashtirish va yangi innovatsion yondashuvlarni yaratishda foydalanishlari mumkin. Yana bir muhim jihat — ChatGPT yordamida ishlab chiqilgan tizimlar odatda foydalanuvchilar bilan interaktiv, intuitiv va qulay ishlaydi. Bu esa zamonaviy dasturiy ta'minot yaratishda juda muhim omil hisoblanadi. Sun'iy intellekt bilan bog'liq etik masalalar ham dolzarb. Ma'lumotlar maxfiyligi, noto'g'ri natijalar ehtimoli, va sun'iy intellektning noto'g'ri ishlatilishi mumkinligi — bu texnologiyaning salbiy tomonlaridan sanaladi. Shuning uchun, dasturchilar nafaqat texnologiyani o'rganishlari, balki uning xavfsiz va foydali ishlatilishini ta'minlash yo'llarini ham o'zlashtirishlari zarur. ChatGPT ishlashda aniq va ravshan promptlar, ya'ni so'rovlar orqali undan to'g'ri natijalarni olish mumkin. Shu sababli "Prompt Engineering" ham so'nggi yillarda muhim yo'nalishlardan biriga aylandi. Sun'iy intellekt — bu kelajak emas, hozirgi zamonning ajralmas qismiga aylangan kuchli vositadir. ChatGPT esa dasturchilar uchun bu kuchdan to'laqonli foydalanish imkonini beruvchi ajoyib platforma hisoblanadi. Ushbu maqolada ChatGPT'ning imkoniyatlari, ishlash mexanizmlari, amaliy qo'llanishi va ehtiyotkorlik talab qilinadigan jihatlari atroflicha yoritiladi.

ChatGPT qanday ishlaydi?

ChatGPT — bu sun'iy intellektga asoslangan matn generatsiyasi modeli bo'lib, u **Generative Pre-trained Transformer** (ya'ni **GPT**) texnologiyasi asosida ishlaydi. Ushbu model OpenAI tomonidan ishlab chiqilgan va aynan tabiiy til bilan ishlash — matnni tushunish, davom ettirish, savollarga javob berish, tarjima qilish, kod yozish kabi vazifalarni bajarishga mo'ljallangan. ChatGPT qanday ishlashini tushunish uchun biz quyidagi asosiy bosqichlarni ko'rib chiqamiz: **transformer arxitekturasini, modelni o'rganish jarayoni, va foydalanuvchi bilan muloqotga kirishish bosqichi.**



GPT modellarining asosi — bu **Transformer** deb nomlangan neyron tarmoq arxitekturasi. Transformer texnologiyasi 2017-yilda Google tadqiqotchilari tomonidan taqdim etilgan bo'lib, u matnni qayta ishlashda ilgari qo'llanilgan rekurrent (RNN) va konvolyutsion (CNN) tarmoqlarga nisbatan sezilarli afzalliklarga ega. Transformer modelining eng muhim komponenti bu — **self-attention mechanism**. U matndagi har bir so'zning boshqa so'zlarga qanday aloqasi borligini aniqlaydi.

Masalan, siz "Olma daraxtdan tushdi, chunki **u** pishgan edi" degan gapni yozsangiz, model "**u**" so'zi "olma"ga tegishli ekanini tushunishi kerak. Transformer'ning self-attention tizimi aynan shunday murakkab bog'lanishlarni aniqlay oladi. Bu esa modelga matn kontekstini chuqur tushunish imkonini beradi.

ChatGPT modeli ikki asosiy bosqichda tayyorlanadi:

Oldindan tayyorlash (Pre-training) - Model avvalo katta hajmdagi matnlar ustida o'qitiladi. Bu matnlar internetdagi maqolalar, kitoblar, veb-sahifalar, kodlar, forumlar va boshqa manbalardan olinadi. O'qitishning maqsadi — modelni umumiy bilim bilan to'ldirish va unga til tuzilishini "his qilish"ni o'rgatish. Masalan, agar model "Bugun ob-havo juda..." so'zini ko'rsa, u ehtimoliy davomini taxmin qilishi kerak: "issiq", "sovuq" yoki "nam" bo'lishi mumkin.

GPT bu bosqichda **unsupervised learning** usulidan foydalanadi, ya'ni tayyor javoblar berilmaydi — model mustaqil o'rganadi. Bu jarayonda milliardlab parametrlarga ega model har bir so'zdan keyin qaysi so'z kelishini bashorat qilishni o'rganadi.

Inson nazorati ostida nozik o'qitish (Fine-tuning with Human Feedback) Bu bosqichda modelga **RLHF (Reinforcement Learning from Human Feedback)** orqali aniqlik va sifat kiritiladi. OpenAI mutaxassislari model bergan javoblarni tahlil qilib, ularni reytinglaydi va yaxshiroq javob berishni o'rgatadi. Bu modelni muloqotga moslashtiradi. Aynan shu bosqichda ChatGPT foydalanuvchi bilan muloqotda "odamga o'xshagan" tarzda gapira boshlaydi — muloyim, tushunarli, muvozanatli va kontekstga mos.



Foydalanuvchi ChatGPT'ga savol berganida yoki buyruq yozganida, bu matn modelga **prompt** sifatida yuboriladi. Model bu matnni analiz qiladi, kontekstni tushunadi, soʻzlar orasidagi mantiqiy bogʻlanishlarni aniqlaydi va eng mos keluvchi javobni yaratadi. Javob koʻpincha ehtimollik asosida tanlanadi — yaʼni model har bir keyingi soʻz uchun yuzlab variantlarni hisoblab chiqadi va eng mosini tanlaydi.

Masalan, siz "Python'da kalkulyator yoz" deb soʻrasangiz, model bu soʻrovni dasturlash kontekstida tahlil qilib, unga mos Python kodini yaratadi. Kod yozish jarayonida model nafaqat sintaksisni, balki amaliyotdagi foydali uslublarni ham hisobga oladi.

ChatGPT quyidagilarda juda samarali ishlaydi:

- ❖ Kod yozish (Python, Java, C++, HTML va h.k.)
- ❖ Kodni tahlil qilish, xatolikni topish
- ❖ Soʻzlar yoki matn tarjimasi
- ❖ Chatbot yaratish
- ❖ Tabiiy tilda soʻrovni kodga aylantirish
- ❖ Maqola, referat, eʼlon yozish

Bundan tashqari, GPT-4 versiyasi rasm, grafik, jadval, kod, matnlar bilan ishlay oladi va ayni vaqtda multimodal (yaʼni koʻp turdagi maʼlumotni tushunadigan) tizimdir. Bu esa undan interfeyslar, mobil ilovalar, AI assistantlar yaratishda keng foydalanish imkonini beradi.

ChatGPT hali ham 100% mukammal emas. U:

- Baʼzida faktik xatolar qiladi.
- Eskirgan yoki oʻzgaruvchan maʼlumotlarga asoslanadi.
- Ehtimollikka asoslangan javoblar tufayli notoʻgʻri yoki noaniq yechim berishi mumkin.
- Koʻp hollarda foydalanuvchi tomonidan toʻgʻri prompt berilmasa, natija notoʻgʻri boʻlishi mumkin.



Shuning uchun ham **prompt engineering** — ya'ni modelga to'g'ri buyruq berish san'ati — muhim soha hisoblanadi.

ChatGPT va Sun'iy Intellekt: Dasturchilar Uchun Imkoniyatlar

Zamonaviy texnologiyalar jadal sur'atlar bilan rivojlanib borayotgan bir davrda, sun'iy intellekt (SI) sohasidagi yutuqlar butun IT sanoatining barcha jabhalariga ta'sir ko'rsatmoqda. Xususan, OpenAI tomonidan ishlab chiqilgan ChatGPT modeli bugungi kunda dasturchilar uchun qulay, samarali va ko'p funksiyali yordamchiga aylangan. Bu model faqatgina matnga asoslangan muloqot qilish imkonini beribgina qolmay, balki real kod yozish, tahlil qilish va muammoni yechish kabi texnik jarayonlarda ham faol foydalanilmoqda.

ChatGPT – bu generativ til modeli bo'lib, uning asosini “Generative Pre-trained Transformer” (GPT) arxitekturasi tashkil etadi. Ushbu model katta hajmdagi matnlar orqali oldindan o'rgatilgan bo'lib, foydalanuvchi tomonidan yuborilgan so'rovga mos javoblar yaratadi. ChatGPT o'zining imkoniyatlari bilan dasturlash sohasida faoliyat yurituvchi mutaxassislar uchun bir nechta yo'nalishda katta yengilliklar yaratadi. Jumladan, kod yozish, xatolarni topish, hujjatlashtirish, algoritmlar yaratish, til o'rganish, yangi texnologiyalarni o'zlashtirish va test sinovlari kabi vazifalarni bajarishda yordam beradi.

Birinchidan, ChatGPT dasturchilarga turli dasturlash tillarida kod yozishda yordam beradi. U Python, Java, C++, JavaScript, HTML, CSS, PHP, Kotlin, Swift, R, SQL va boshqa ko'plab dasturlash tillarida to'g'ri sintaksisda kod yaratadi. Foydalanuvchi modelga oddiy tushunchada topshiriq berganida, ChatGPT bu topshiriqni algoritmik tahlil qiladi va unga mos kod namunasini yaratadi. Bu holat, ayniqsa, yangi o'rganuvchilar uchun muhim ahamiyatga ega, chunki ular real kodlarni tahlil qilish orqali amaliy ko'nikma hosil qilishadi.

Ikkinchidan, ChatGPT yordamida mavjud koddagi xatolarni aniqlash va tuzatish mumkin. Dasturchi o'zining ishlayotgan kodini modelga yuboradi va undan “Bu kod nima uchun ishlaymapti?” deb so'raydi. Model esa kodni tahlil qilib, sintaktik yoki



mantiqiy xatolarni ko'rsatadi va tuzatish bo'yicha tavsiyalar beradi. Bu jarayon vaqtni tejaydi va muammoni ancha tez yechishga yordam beradi.

Uchinchidan, ChatGPT algoritmik muammolarni yechishda ham dasturchilarga katta yordamini ko'rsatmoqda. Masalan, foydalanuvchi "Bubble sort algoritmini Java tilida yozib ber" yoki "Fibonacci sonlarini rekursiv usulda chiqaring" degan so'rov bilan murojaat qilsa, model muammoning yechimini tushuntiradi va kod bilan birga izoh beradi. Bu esa o'rganuvchilarga nafaqat kodni ko'rish, balki algoritmik tafakkurni rivojlantirish imkonini ham yaratadi.

To'rtinchidan, dasturchilar uchun dokumentatsiya yozish, texnik topshiriq hujjatlari tayyorlash va dasturiy loyihalarning texnik tafsilotlarini ifodalash ko'p vaqt talab qiladigan jarayondir. ChatGPT bu borada ham yordamchi vosita sifatida xizmat qiladi. Foydalanuvchi modelga dastur funksiyalarini aytadi, model esa ularni texnik tarzda bayon qilib beradi. Bu loyihalar hujjatlarini tizimli va tushunarli tarzda tayyorlashga yordam beradi.

Beshinchidan, dasturchilar ko'pincha yangi texnologiyalar, kutubxonalar yoki framework'larni o'rganishga ehtiyoj sezishadi. ChatGPT bu borada qisqa, tushunarli, va tezkor ma'lumotlar berishga qodir. Masalan, foydalanuvchi "React bilan komponent qanday yaratiladi?" yoki "TensorFlow'da model qanday o'rgatiladi?" degan savollar bilan murojaat qilsa, model tegishli javobni, kod misollari bilan birga, aniq va tushunarli qilib beradi.

ChatGPT dasturchilarga test sinovlarini yozishda ham yordam beradi. Unit test, integration test yoki regression test kabi jarayonlar dasturiy ta'minot sifatini tekshirishda muhim rol o'ynaydi. Model dasturchining yozgan funksiyasi asosida test kodlarini yaratib bera oladi. Bu esa dasturiy mahsulotlar xavfsizligini ta'minlash va xatolarning oldini olishda muhim ahamiyat kasb etadi. Yana bir muhim jihat – ChatGPT yordamida ishlab chiqilayotgan dastur interfeysini takomillashtirish mumkin. Masalan, foydalanuvchi “Menga foydalanuvchi uchun qulay HTML sahifa yaratib ber” deb murojaat qilganda,



model zamonaviy dizayn, foydalanuvchi tajribasi (UX) va foydalanuvchi interfeysi (UI) tamoyillariga asoslangan kod namunalarini taqdim etadi.

ChatGPT yordamida dasturchilar o'zlarining kommunikatsion ko'nikmalarini ham rivojlantirishlari mumkin. Dasturchi loyiha taqdimoti, ish jarayoni bo'yicha hisobot yoki muloqot yozishmalari kerak bo'lsa, model unga rasmiy, tushunarli va stilistik jihatdan to'g'ri matnlar tuzib beradi. Bu esa jamoaviy ishlarda, loyiha menejmentida va mijozlar bilan aloqada bo'lishda qo'l keladi.

ChatGPT dasturchilar uchun ko'p qirrali, kuchli va ishonchli vositaga aylangan. U nafaqat kod yozishda, balki algoritmik tafakkurni shakllantirish, til o'rganish, test sinovlarini yozish, hujjatlar tayyorlash, yangi texnologiyalarni o'zlashtirish va hatto dasturiy loyihalarni boshqarishda ham muhim o'rin tutadi. Dasturchilar bu modeldan maqsadli, mas'uliyatli va ijodiy foydalanish orqali o'z ish unumdorligini bir necha barobarga oshirishlari mumkin.

Amaliy misollar

1. Kod yozish (Python kalkulyator)

Vaziyat: Boshlovchi dasturchi oddiy kalkulyator dasturini Pythonda yozmoqchi, lekin sintaksisda xatolarga yo'l qo'yadi.

So'rov:

“Menga Pythonda oddiy kalkulyator dasturini yozib ber.”

ChatGPT javobi:

```
def calculator():  
  
    print("Amallar: +, -, *, /")  
  
    amal = input("Amalni kiriting: ")  
  
    a = float(input("Birinchi son: "))  
  
    b = float(input("Ikkinchi son: "))
```



```
if amal == '+':  
    print(f"Natija: {a + b}")  
elif amal == '-':  
    print(f"Natija: {a - b}")  
elif amal == '*':  
    print(f"Natija: {a * b}")  
elif amal == '/':  
    print(f"Natija: {a / b}")  
else:  
    print("Noto'g'ri amal kiritildi.")
```

calculator()

2.Xatoni topish (JavaScript kodida xatolik)

Vaziyat: JavaScriptdagi kod ishlamayapti. Dasturchi sababini tushunmayapti.

Kod (xato bilan):

```
let num = prompt("Son kiriting");  
  
if (num = 5) {  
    alert("Son 5 ga teng");  
}
```

ChatGPT tahlili:



"if (num = 5)" - bu yerda "=" belgisi qiymat berish uchun ishlatilgan. Solishtirish uchun "==" yoki "===" ishlatilishi kerak.

To'g'rilangan kod:

```
let num = prompt("Son kiriting");  
  
if (num == 5) {  
    alert("Son 5 ga teng");  
}
```

3. Algoritm yechimini tushuntirish (Fibonacci sonlari)**So'rov:**

“Menga Pythonda Fibonacci sonlarini chiqaradigan funksiya yozib ber va qanday ishlashini tushuntir.”

Javob:

```
def fibonacci(n):  
    a, b = 0, 1  
  
    for i in range(n):  
        print(a, end=" ")  
  
        a, b = b, a + b
```

fibonacci(10)

Izoh: Har bir son avvalgi ikki sonning yig'indisidir. Bu funksiya 10 ta Fibonacci sonini chiqaradi.



4. HTML sahifa yaratish (frontend misoli)

So‘rov:

“Oddiy aloqa formasi HTMLda yozib ber.”

Javob:

```
<form action="/submit" method="post">  
  
  <label>Ismingiz:</label><br>  
  
  <input type="text" name="name"><br><br>  
  
  <label>Xabar:</label><br>  
  
  <textarea name="message"></textarea><br><br>  
  
  <input type="submit" value="Yuborish">  
  
</form>
```

5. Test yozish (Unit test – Python)

So‘rov:

“Pythondagi funktsiyani test qilish uchun unittest yozib ber.”

Funktsiya:

```
def add(x, y):  
  
    return x + y
```

Unittest:

```
import unittest
```



```
class TestAddFunction(unittest.TestCase):
```

```
    def test_add(self):
```

```
        self.assertEqual(add(2, 3), 5)
```

```
        self.assertEqual(add(-1, 1), 0)
```

```
if __name__ == '__main__':
```

```
    unittest.main()
```

6. Yangi texnologiyani tushuntirish (React komponent)

So'rov:

“Menga oddiy React komponentini yozib ber.”

Javob:

```
import React from 'react';
```

```
function Salom() {
```

```
    return (
```

```
        <div>
```

```
            <h1>Salom, dunyo!</h1>
```

```
        </div>
```

```
    );
```

```
}
```




export default Salom;

7. API bilan ishlash (OpenWeatherMap)

So'rov:

“OpenWeatherMap API bilan ob-havo ma'lumotini Python'da olib ber.”

Javob:

```
import requests
```

```
api_key = "API_KALITINGIZ"
```

```
city = "Tashkent"
```

```
url =
```

```
f"https://api.openweathermap.org/data/2.5/weather?q={city}&appid={api_key}&units=metric"
```

```
response = requests.get(url)
```

```
data = response.json()
```

```
print(f"{city} uchun havo harorati: {data['main']['temp']} °C")
```

8. SQL so'rovini tuzish (ma'lumotlar bazasi)

So'rov:

“Yosh > 25 bo'lgan foydalanuvchilarni tanlab beradigan SQL so'rovi yoz.”

Javob:

```
SELECT * FROM foydalanuvchilar WHERE yosh > 25;
```

**Foydalanilgan adabiyotlar ro'yxati:**

1. OpenAI. (2023). *ChatGPT: Optimizing Language Models for Dialogue*. <https://openai.com/chatgpt>
2. Brown, T., Mann, B., Ryder, N., et al. (2020). *Language Models are Few-Shot Learners*. arXiv:2005.14165.
3. Russell, S., & Norvig, P. (2021). *Artificial Intelligence: A Modern Approach* (4th ed.). Pearson.
4. Chollet, F. (2017). *Deep Learning with Python*. Manning Publications.
5. Goodfellow, I., Bengio, Y., & Courville, A. (2016). *Deep Learning*. MIT Press.
6. Vaswani, A., Shazeer, N., Parmar, N., et al. (2017). *Attention is All You Need*. NeurIPS.
7. Zeller, M. (2023). *How AI is Transforming Software Development*. IEEE Software Magazine.
8. TIOBE Index for June 2025. (2025). *Most Popular Programming Languages*. <https://www.tiobe.com/tiobe-index>
9. GitHub Copilot Documentation. (2024). <https://docs.github.com/copilot>
10. Karimov, A. (2022). *Sun'iy intellekt asoslari*. Toshkent: INNOVATION nashriyoti.
11. O'zbekiston Respublikasi Raqamli texnologiyalar vazirligi. (2023). *Sun'iy intellekt strategiyasi – 2030*. <https://digital.gov.uz>
12. BERT, GPT va Transformer modellar sharhi – TowardsDataScience.com. (2022). <https://towardsdatascience.com>
13. Ng, A. (2023). *AI for Everyone*. Coursera kurs material. <https://www.coursera.org>
14. Google AI Blog. (2023). *The Future of Generative AI*. <https://ai.googleblog.com>