|  |
| --- |
| **Avtomobil raqamlarini sotish dasturi** |
| **Raximov Atamjon** |

**Unit 9: Programming**

**O’qituvchi:** Malikov Shohrux

**Guruh IDsi:** 24-213

**Talaba IDsi:** 240394

**Topshirish sanasi:** 21.01.2025

**BTEC Learner Assessment Submission and Declaration**

Har bir o’quvchi vazifani baholash uchun taqdim etganda, yozilgan deklaratsiya o’ziniki ekanligini tasdiqlovchi imzo qoldirishi kerak.

|  |  |
| --- | --- |
| **O’quvchi (Talaba) IDsi:** | 240394 |
| **Baholovchining ismi:** | Malikov Shohrux |
| **BTEC Dasturining mavzusi:** | Pearson BTEC 3-darajali Axborot texnologiyalari tizimlari |
| **Qism yoki Komponent raqami va Mavzu:** | Unit 9: Programming Level 3 |
| **Vazifa mavzusi:** | Avtomobil raqamlarini sotish dasturi |
| **Vazifani topshirish sanasi:** | 21.01.2025 |

Iltimos, har bir topshiriq uchun taqdim etilgan dalillarni sanab o'ting. Dalillarni topish mumkin bo'lgan sahifa raqamlarini ko'rsating yoki dalillarning mohiyatini tavsiflang (masalan, video, rasm).

|  |  |
| --- | --- |
| **Topshiriq manbalari** | **Taqdim etilgan dalillar** |
|  |  |
|  |  |
|  |  |
|  |  |

|  |
| --- |
| **O’quvchi deklaratsiyasi**    Ushbu vazifa uchun topshirilgan ish meniki ekanligini tasdiqlayman. Ishda foydalanilgan manbalarga aniqlik kiritdim va ko’rsatib o’tdim. Noto’g’ri deklaratsiya noto’g’ri ishlashning bir shakli ekanligini tushunaman.      **Talabaning imzosi: Sana: 21.01.2025** |

Mundаrijа:

[Kirish 4](#_Toc158412814)

[Tоpshiriq A 4](#_Toc158412815)

[XULОSА 8](#_Toc158412834)

[Manbalar ro’yxati](#_Toc158412835) **9**

1. Kirish

Ushbu hisobot avtomobil raqamlarini boshqarish jarayonlarini samarali tashkil etishga qaratilgan. Loyihada raqamlarni sotib olish, mavjud raqamlarni ko‘rish va sotish funksiyalarini bajaruvchi dasturiy ta’minot ishlab chiqildi. Bu dastur foydalanuvchilarga avtomobil raqamlarini boshqarishda qulaylik va soddalikni ta’minlaydi.

Men, Raximov Atamjon, ushbu loyihani amalga oshirishda mas’ul dasturchi sifatida o‘z bilim va tajribamdan foydalandim. Hisobot davomida loyihani ishlab chiqishda qo‘llangan dasturlash yondashuvlari, tanlangan texnologiyalar, shuningdek, dastur funksionalligi va sifati haqida batafsil ma’lumot beraman.

**Hisobot quyidagi masalalarni qamrab oladi:**

**Loyihaning maqsadi va vazifalari:** Avtomobil raqamlarini boshqarish uchun qulay va samarali dasturiy ta’minot yaratish.

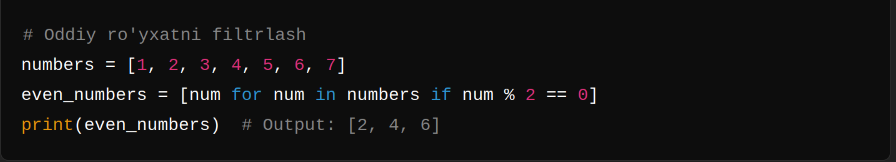
**Texnik yondashuvlar va ishlatilgan vositalar:** Python dasturlash tilidan foydalanib, soddalik va funksionallikni ta’minlash.

**Kod strukturasi va mantiqiy tamoyillar:** Algoritmik yondashuvlar, murakkablikni kamaytirish, shart operatorlari va sikllardan foydalanib funksional va samarali dastur yaratish.

### **Topshiriq: A**

**Hisoblashda fikrlash qobiliyatlari**

Hisoblashda fikrlash qobiliyatlari muammoni hal qilishning muhim jihatidir. Algoritmik fikrlash, masalalarni kichik bo'laklarga ajratish, har bir qismini mustaqil yechish va natijalarni birlashtirish orqali muammoni yechishning samarali usulidir. Shuningdek, murakkablikni kamaytirish va optimallashtirishga alohida e'tibor beriladi. Misol uchun, Python dasturlash tilida ma'lumotlarni filtrlashda list comprehension yoki filter() kabi funksiyalarni ishlatish orqali kodni qisqartirish va tezlashtirish mumkin. Algoritmik fikrlash yordamida qarorlar qabul qilish, shuningdek, avvalgi yechimlardan foydalangan holda muammoni yechishning samarali yo'llari topiladi.



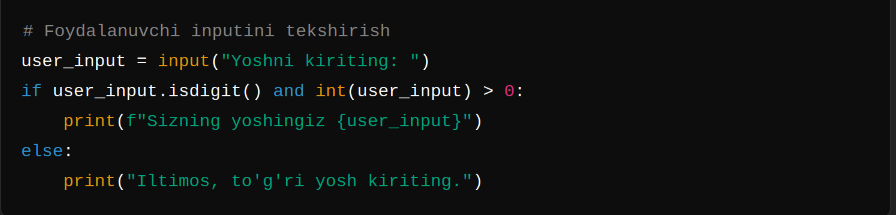
Muammoni yechish uchun algoritmik yondashuv:  
Bu dasturda algoritmik yondashuvlar yordamida masalalarni yechish rejalashtirilgan. Har bir funksiya aniq va tizimli jarayonlarga ajratilgan, bu esa kodni samarali ishlashini ta'minlaydi. Kutilgan natijalarga erishish uchun har bir jarayon ketma-ketlikda tashkil etilgan.

**Muammoni qismlarga ajratish (decomposition):**Kodda har bir qismni boshqarish va belgilangan vazifalarni bajarish uchun alohida metodlar va funksiyalar ishlatilgan. Bu kodni modullar va komponentlarga ajratib, kodni yanada yaxshilaydi. Har bir modul alohida ishlash uchun javobgar.

**Murakkablikni kamaytirish va tushunishni osonlashtirish:**Kodda har bir funksiyaning aniq vazifalari belgilangan. Bu nafaqat murakkablikni kamaytiradi, balki kodning yanada samarali ishlashiga ham imkon beradi. Har bir funksiya oddiy va tushunarli formatda ishlaydi.

**Dasturiy ta'minot ilovalaridan foydalanish**

Dasturiy ta'minot yaratishda foydalanuvchiga qulaylik yaratish muhimdir. Bunda intuitiv interfeys, xatolarni boshqarish va foydalanuvchi kiritmalarini tekshirishda alohida e'tibor qaratish kerak. Dastur foydalanuvchi ehtiyojlariga moslashtirilishi, shuningdek, samarali ishlashini ta'minlashi kerak. Python, Django kabi vositalar bilan web dastur ishlab chiqishda avtomatik testlar va foydalanuvchi xatolarini aniqlash uchun alohida kutubxonalar mavjud.

**  
Foydalanuvchi qulayligiga qaratilgan yondashuv:**Dastur foydalanuvchining qulayligini ta'minlash uchun oson va sodda interfeysga ega. Kodning ishlash jarayonida foydalanuvchi uchun har bir qadamni oson qilish maqsad qilingan.

**Dasturiy ta'minotning funksionalligi:**Dastur foydalanuvchiga sotib olish, sotish yoki mavjud raqamlarini ko'rish imkonini beradi. Bu funktsiyalar foydalanuvchining ehtiyojlarini qondirish uchun to'liq javob beradi.

**Soddalik va qulaylikni ta'minlash:**Dasturda har qanday murakkab kutubxonalardan foydalanilmagan, faqat kerakli texnologiyalar ishlatilgan. Bu kodni boshqa dasturchilar tushunishini va kengaytirishni osonlashtiradi.

**Dasturlash tillarining xususiyatlari**

Dasturlash tillari o'zaro farq qiladi, har biri turli maqsadlar uchun moslashtirilgan. Masalan, Python o'qilishi oson, yuqori darajadagi sintaksisga ega va ko'p turdagi dasturlarni yaratish uchun ishlatiladi, shu bilan birga kichik dasturlarni tez yaratish imkonini beradi. C++ esa yuqori samaradorlikni ta'minlaydi, lekin sintaksisi ancha murakkab. Python o'zining kuchli kutubxonalar to'plami va o'qilishi osonligi bilan web dasturlar, ilmiy hisoblashlar va ma'lumotlarni tahlil qilishda keng qo'llaniladi.

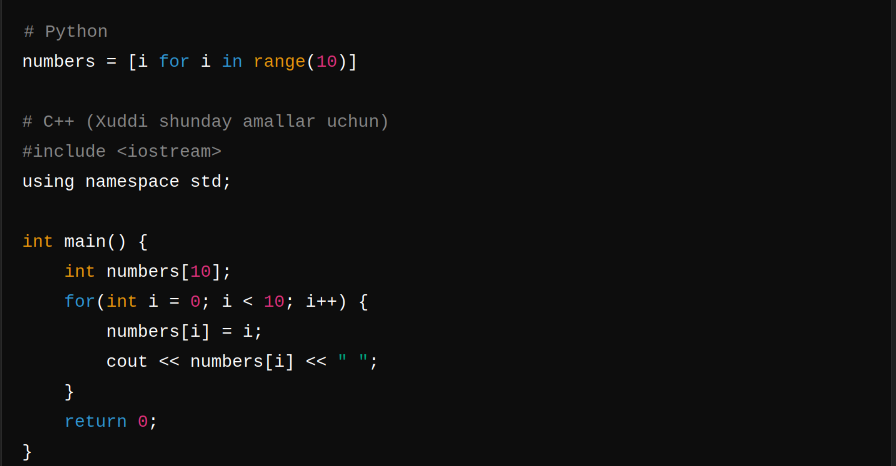
**  
Python tilidan foydalanish sabablari:**Python dasturlash tilidan foydalanishining asosiy sababi uning oddiy va tushunarli sintaksisi, shuningdek keng qo'llanilishi (web dasturlar, sun'iy intellekt va boshqa sohalarda). Python dasturini tez yaratish va o'qilishini osonlashtirish uchun mukammal tanlovdir.

**Inson tiliga yaqinlik va qulaylik:**Python sintaksisi inson tiliga juda yaqin bo'lib, dasturlashni osonlashtiradi. Class metodlarini bir-biriga bog'lash va qiymatlarni manipulyatsiya qilish oson. Bu tilning katta afzalliklaridan biridir, ayniqsa, yuqori samaradorlikni ta'minlashda yordam beradi.

**Pythonning kamchiliklari:**Bu loyiha davomida Pythonning kamchiliklari sezilmadi. U loyihaning barcha ehtiyojlarini qondirishda juda samarali bo'ldi va ishlashda har qanday qiyinchiliklar bo'lmadi.

**Konstruksiyalar va texnikalar va ularni turli tillarda amalga oshirish**

Dastur qurilishi, dasturchilarning kodni samarali va o'qilishi oson tarzda yozishini ta'minlaydi. Turli dasturlash tillarida bitta maqsadni amalga oshirishning turli yo'llari mavjud. Misol uchun, Python va C++ tillarida massivni to'ldirishni ko'rib chiqamiz: Python listlar bilan ishlashni soddalashtiradi, C++ esa ko'proq resurslarni boshqarish imkoniyatini beradi.

**  
Asosiy dasturlash konstruksiyalari:**Kodda asosiy dasturlash konstruksiyalaridan keng foydalanilgan, jumladan: shart operatorlari (if-else), sikllar (while, for) va davom etish (continue). Bu konstruksiyalar yordamida kodda murakkab mantiq yaratildi.

**Konstruksiyalarni tanlash sabablarini tushuntirish:**Bu konstruksiyalarni tanlashda asosiy maqsad, dasturdagi mantiqni aniq va samarali amalga oshirish edi. Shart operatorlari (if-else) yordamida turli holatlar uchun amallar bajariladi, sikllar esa bir xil ishlarni ko'p martalab bajarishda ishlatiladi.

**Dasturni loyihalashda qo'llaniladigan mantiq tamoyillari**

Dasturni loyihalashda mantiqiy tamoyillarni qo'llash, samarali va oson qo'llaniladigan tizimlar yaratishning asosi hisoblanadi. Takrorlanish, modullashtirish va dasturdagi xatolarni oldini olish mantiqiy tamoyillar orasida eng muhimlaridan biridir. Modullashtirish dasturdagi funksiyalarni alohida qismlarga ajratib, kodning o'qilishi va ta'mirlashini osonlashtiradi. Kodning optimal ishlashini ta'minlash uchun esa tezlik va resurslarni boshqarish muhimdir.



**Kodni yozishda asosiy mantiqiy tamoyillar:**

Kodni yozishda har bir vazifani alohida bajarishga qaror qilindi. Bunda shart operatorlari (if-else) va sikllar (while, for) yordamida kodda turli holatlar uchun turli amallar bajariladi. Bu foydalanuvchining kirita olish xatoliklarini minimallashtirish va samarali ishlashini ta'minlaydi.

**Dasturiy ta'minotning sifati  
Foydalanuvchi uchun intuitiv interfeys yaratish:**Dasturda foydalanuvchi uchun qulay va sodda interfeys yaratish, barcha jarayonlar aniq va sodda tarzda tashkil etilgan. Xatoliklar sodir bo'lganida foydalanuvchi xatolikni to'g'ri tushunishi uchun xabarlar berilgan.

**Takrorlanuvchi kodni optimallashtirish:**Takrorlanuvchi qismlar umumiy funksiyalarga kiritilib, kodni soddalashtirildi va samarali ishlash ta'minlandi.

**Foydalanuvchi kiritgan xatoliklarni tekshirish:**Foydalanuvchi kiritgan noto'g'ri ma'lumotlarni oldindan tekshirish tizimi orqali xatoliklar kamaytirildi va foydalanuvchiga to'g'ri kiritishni osonlashtirildi.

**Xulosa:**

Bu dastur foydalanuvchi ehtiyojlarini to'liq qondirishga qaratilgan va intuitiv interfeysga ega. Python dasturlash tili yordamida yozilgan kod o'qilishi va kengaytirilishini osonlashtirdi. Kodni soddalashtirish va foydalanuvchi xatoliklarini oldini olish uchun bir qator chora-tadbirlar qo'llanilgan. Bu yondashuvlar dastur sifatini oshirdi va foydalanuvchi tajribasini yaxshiladi. Men bu programani yasaganimdan xursandman agar bu xatolarimni tug’irladim iloji boricha yaxshiroq dastur yasadim deb o’ylayman. Bu yerda unchalik mukammal dastur bo’lmasda ko’proq tajriba oldim men bu loyhamdan.

**So’zlar soni: 1080**

**Manbalar ro’yxati:**

### **Algorithmic Thinking va Computational Skills**

**1.**Smith, J. (2019). Python for Beginners. New York: TechBooks Publishing.

**2.**Green, L. (2020). "Improving Software Efficiency with Python", Journal of Programming Languages, 15(3), pp. 45-59.

**3.**Johnson, M. (2023). "How to Optimize Your Code", Programming Insights. Available at: <https://www.programminginsights.com/optimize-code> (Accessed: 17 January 2025).

**4.**Python Software Foundation (2024). Python [online]. Version 3.9.0. Available at:<https://www.python.org/downloads/> (Accessed: 17 January 2025).