

Programming

Yo'lchiyev Abdulqodir

4-qism: Programming

O'qituvchi: Azamat Hamrayev

Guruh IDsi: 23-218

Talaba IDsi: 230641

Topshirish sanasi: 10.01.2024

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: | 230641 |
| Baholovchining ismi: | Azamat Hamrayev |
| BTEC Dasturining mavzusi: | Pearson BTEC Programming |
| Qism yoki Komponent raqami va Mavzu: | 4-qism: Programming |
| Vazifani topshirish sanasi: | 10.01.2024 |

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).

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: 10.01.2024

1.0. KIRISH

Men Yo'lchiyev Abdulqodir restaurant uchun dastur yasadim bu dastur restorandagi joylarni online bron qilish uchun katta yordam beradi va kollejlarda uchun baholash sistemasini o'rganib chiqdim.

2. TOPSHIRIQ 1

RESTORANNI BRON QILISH TIZIMI NIMA?

Restoranni band qilish tizimi barcha buyurtma laringizni raqamli tarzda boshqarishga yordam beradi, shunda siz o'sha qo'rqinchli shovqinlarni kamaytirishingiz va barcha mavjud jadvallaringizni maksimal darajada oshirishingizga ishonch hosil qilishingiz mumkin

• DASTURNING KODINI TAHLILI

```
Import java.util.HashMap;
```

```
Import java.util.Map;
```

```
Import java.util.Scanner;
```

Birinchi bo'lib bizga zarur kutubxonalarni o'zimizga chaqirib olamiz **HASHMAP** zahira ma'lumotlar tuzilmasi; **MAP** interfeys uchun chaqirib olindi ;**SCANNER** foydalanuvchi uchun;

Keyin class ochi olamiz **"CLASS RESTAURANTRESERVATIONSYSTEM{"**VA

"PRIVATE MAP<STRING, STRING[]>RESERVATIONS;bu qatorda yozilgan kod Map

turidagi shaxsiy o'zgaruvchini elon qiladi.Keyingi qatorda dasturda foydalanuvchi uchun ko'rinadigan gaplarni yozib olamiz

```
public void displayMenu() {
    System.out.println("\nRestaurant Table Reservation System");
    System.out.println("1. Make a Reservation");
    System.out.println("2. Update a Reservation");
    System.out.println("3. Cancel a Reservation");
    System.out.println("4. View Reservations");
    System.out.println("5. Exit");
}

public void makeReservation() {
    Scanner scanner = new Scanner(System.in);
    System.out.print("Enter customer name: ");
    String name = scanner.nextLine();
    System.out.print("Enter reservation date (YYYY-MM-DD): ");
    String date = scanner.nextLine();
    System.out.print("Enter reservation time (HH:MM): ");
    String time = scanner.nextLine();
    System.out.print("Enter number of people: ");
    String people = scanner.nextLine();
    String key = date + "," + time;

    if (reservations.containsKey(key)) {
        System.out.println("A reservation already exists at this time. Please choose another time.");
    } else {
        reservations.put(key, new String[]{name, people});
        System.out.printf("Reservation made for %s on %s at %s for %s people.%n", name, date, time, people);
    }
}
```

Bu usul **bron qilish** jarayonini boshqaradi. U foydalanuvchidan mijoz nomi, bandlov sanasi, vaqti va odamlar soni kabi ma'lumotlarni yig'ish uchun **Scanner** dan foydalanadi.

```

public void updateReservation() {
    Scanner scanner = new Scanner(System.in);
    System.out.print("Enter reservation date (YYYY-MM-DD) to update: ");
    String date = scanner.nextLine();
    System.out.print("Enter reservation time (HH:MM) to update: ");
    String time = scanner.nextLine();
    String key = date + "," + time;

    if (!reservations.containsKey(key)) {
        System.out.println("Reservation not found.");
    } else {
        System.out.print("Enter new reservation date (YYYY-MM-DD): ");
        String newDate = scanner.nextLine();
        System.out.print("Enter new reservation time (HH:MM): ");
        String newTime = scanner.nextLine();
        String newKey = newDate + "," + newTime;

        if (reservations.containsKey(newKey)) {
            System.out.println("A reservation already exists at this time. Please choose another
time.");
        } else {
            reservations.put(newKey, reservations.remove(key));
            System.out.printf("Reservation updated to %s at %s.%n", newDate, newTime);
        }
    }
}

```

updateReservation funksiyasi **mavjud bronni yangilash** jarayonini boshqaradi. U foydalanuvchidan bron qilish uchun asl sana va vaqtni hamda yangi sana va vaqtni kiritishni taklif qiladi.

```

public void cancelReservation() {
    Scanner scanner = new Scanner(System.in);
    System.out.print("Enter reservation date (YYYY-MM-DD) to cancel: ");
    String date = scanner.nextLine();
    System.out.print("Enter reservation time (HH:MM) to cancel: ");
    String time = scanner.nextLine();
    String key = date + "," + time;

    if (!reservations.containsKey(key)) {
        System.out.println("Reservation not found.");
    } else {
        reservations.remove(key);
        System.out.println("Reservation cancelled.");
    }
}

```

cancelReservation funksiyasi mavjud **bronni bekor qilish** jarayonini boshqaradi. U foydalanuvchidan bekor qilinadigan bron sanasi va vaqtini kiritishni taklif qiladi.

```

while (true) {
    system.displayMenu();
    System.out.print("Choose an option: ");
    String choice = scanner.nextLine();

    switch (choice) {
        case "1":
            system.makeReservation();
            break;
        case "2":
            system.updateReservation();
            break;
        case "3":

```

```

        system.cancelReservation();
        break;
    case "4":
        system.viewReservations();
        break;
    case "5":
        System.out.println("Exiting system. Goodbye!");
        System.exit(0);
    default:
        System.out.println("Invalid choice. Please try again.");
    }
}
}
}

```

Bu qismda while loopidan foydalanilgan bu foydalanuvchi uchun cheksiz sikni yaratib beradi. Mijoz bir vaqtni o'zida bir neshta amalni amaga oshirish uchun yordam beradi.

Foydalanuvchi interfeysi (UI):

Hisoblash fikrlash: Intuitiv va foydalanuvchilarga qulay interfeysni loyihalash

foydalanuvchi o'zaro ta'sirini mantiqiy bosqichlarga bo'lishni talab qiladi. Bu foydalanuvchilarning tizim bilan qanday munosabatda bo'lishini tushunish, ularning ehtiyojlarini oldindan bilish va uzluksiz tajribani taqdim etishni o'z ichiga oladi. Dasturlash tamoyillari: UI ni amalga oshirish grafik elementlarni kodlash, foydalanuvchi kiritishlari bilan ishlash va umumiy javob berishni ta'minlashni o'z ichiga oladi. Hodisalarga asoslangan dasturlash kabi printsiplar o'ynaydi, bu erda tugmani bosish kabi harakatlar muayyan funktsiyalarni ishga tushiradi.

MA'LUMOTLAR BAZASINI BOSHQARISH:

Hisoblash fikrlash: Rezervasyonlarni boshqarish dinamik ma'lumotlar to'plami bilan ishlashni o'z ichiga oladi. Hisoblash fikrlash rezervatsiyalar qanday saqlanishi, olinishi va yangilanishini samarali tarzda tuzishga yordam beradi. Dasturlash tamoyillari: Ma'lumotlar bazasi tizimini amalga oshirish zahira ma'lumotlarini yaratish, o'qish, yangilash va o'chirish (CRUD) uchun ma'lumotlar bazasi tillaridan (masalan, SQL) foydalanishni o'z ichiga oladi. Oddiylashtirish va indekslash kabi tushunchalar ma'lumotlarni samarali boshqarishga yordam beradi.

ALGORITMIK MANTIQ:

Hisoblash fikrlash: tizim jadvallarni qanday taqsimlashi, band qilish mojarolarini hal qilish va o'tirish tartibini optimallashtirish algoritmik fikrlashni talab qiladi. Bu murakkab muammolarni boshqarilishi mumkin bo'lgan bosqichlarga ajratishni o'z ichiga oladi. Dasturlash tamoyillari: Algoritmilar kodga tarjima qilinadi. Misol uchun, o'tirish algoritmi o'rindiqlarni ajratish uchun stol o'lchamlari, mavjud vaqtlar va mijozlarning afzalliklarini hisobga olishi mumkin. Bu erda shartli bayonotlar va tsikllar kabi dasturlash tamoyillari o'ynaydi.

XATOLARNI QAYTA ISHLASH VA TEKSHIRISH:

Hisoblash fikrlash: noto'g'ri kiritilgan ma'lumotlar yoki qarama-qarshi rezervatsiyalar kabi potentsial xatolarni oldindan bilish tizim ishlamay qolishi mumkin bo'lgan turli stsenariylarni ko'rib chiqishni talab qiladi. Dasturlash tamoyillari: Xatolarni qayta ishlash mexanizmlarini amalga oshirish kodda shartli bayonotlar, istisnolarni qayta ishlash va tekshirish tekshiruvlaridan

foydalanishni o'z ichiga oladi. Bu tizimning kutilmagan vaziyatlarga to'g'ri javob berishini ta'minlaydi.

XAVFSIZLIK CHORALARI:

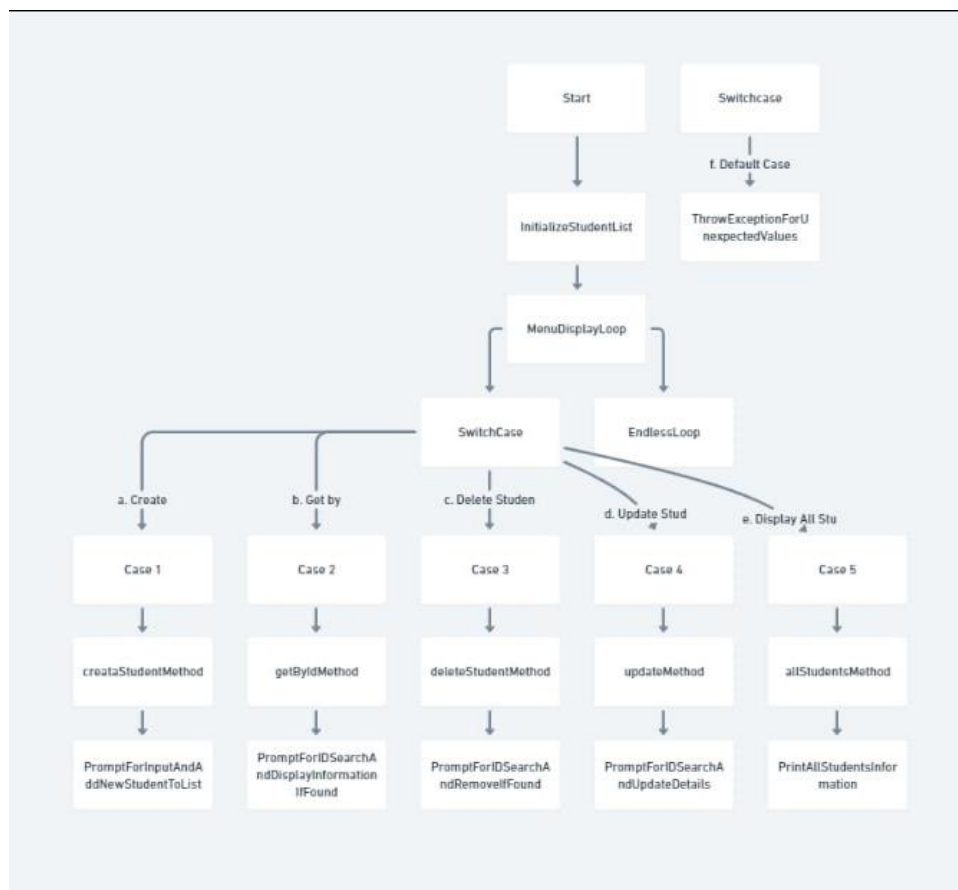
Hisoblash fikrlash: zahira ma'lumotlariga ruxsatsiz kirish kabi potentsial xavfsizlik tahdidlarini aniqlash juda muhimdir. Hisoblash fikrlash nozik ma'lumotlarni himoya qilish strategiyalarini ishlab chiqishda yordam beradi. Dasturlash tamoyillari: Xavfsizlik choralari amalga oshirish autentifikatsiya va avtorizatsiya jarayonlarini kodlashni, maxfiy ma'lumotlarni shifrlashni va xavfsizlik protokollarini muntazam yangilashni o'z ichiga oladi.

TASHQI XIZMATLAR BILAN INTEGRATSIYA:

Hisoblash fikrlash: Onlayn to'lov tizimlari yoki bildirishnoma xizmatlari kabi tashqi xizmatlar bilan integratsiya qilish restoranni bron qilish tizimining ushbu tashqi komponentlar bilan qanday ishlashini tushunishni talab qiladi.

TOPSHIRIQ 2

B.P4. MIJOZ TALABLARIGA MOS KELADIGAN KOMPYUTER DASTURI UCHUN DIZAYN YARATISH.



Yozilgan kodni dizayni shunday qilindi

DIZAYNNING YOZMA KO'RINISHI

Boshlash

Dastur asosiy usulda boshlanadi.

CRUDStudent misolini yaratadi.

Boshlash usulini chaqiradi.

Talabalar ro'yxatini boshlas

addStudents usuli oldindan belgilangan qiymatlarga ega talabalar ro'yxatini ishga tushiradi.

Menyuni ko'rsatish davri

Menyuni ko'rsatish uchun cheksiz tsiklni kiritish

MENYU usuli yordamida menyuni ko'rsatish.

Foydalanuvchi raqam kiritadi.

O'chirish qutisi

Foydalanuvchi kiritgan ma'lumotlarga asoslangan holda almashtirish.

a. Talaba yaratish (1-holat)

creataStudent usulini chaqiradi.

Kirishni taklif qiladi va ro'yxatga yangi talaba qo'shadi.

b. ID orqali oling (2-holat)

Qo'ng'iroqlar getById usuli.

ID so'raydi, ro'yxatni qidiradi va topilsa, talaba ma'lumotlarini ko'rsatadi.

c. Talabani o'chirish (3-holat)

Qo'ng'iroqlarni o'chirishStudent usuli.

ID so'raydi, ro'yxatni qidiradi va topilsa, talabani o'chirib tashlaydi.

d. Talabani yangilash (4-holat)

Qo'ng'iroqlarni yangilash usuli.. ID so'raydi, ro'yxatni qidiradi va foydalanuvchiga talabani ismi va familiyasini yangilash imkonini beradi.

e. Barcha talabalarni ko'rsatish (5-holat)

AllStudent usulini chaqiradi.

Ro'yxatdagi barcha talabalar uchun ma'lumotlarni chop etadi.

f. Standart holat

Kutilmagan qiymatlar uchun istisno qiladi.

Cheksiz tsikl

Dastur qo'lda to'xtatilgunga qadar menyuni ko'rsatishda davom etadi.

C.P6. MIJOZ TALABLARIGA JAVOB BERADIGAN KOMPYUTER DASTURINI YARATISH

TUZULGAN PSEUDACODENI TAHLIL QILSH:

• TALABALAR GROUPALARNI BOXOLASH DASTURIY TAMINOTI:

```
import java.util.LinkedList;
```

```
import java.util.List;
```

```
import java.util.Scanner;
```

Ushubu dastur uchun kerak boladigan class larni import qilsh.

List interfeys, LinkedList Talabalarni saqlash uchun

Scanner foydalanuvchi kiritishu uchun

```
public class CRUDStudent {
```

BU code CRUDStudent nomli classi .

```
public static Scanner scanner = new Scanner(System.in);
```

```
public static Student student = new Student();
```

```
List<Student> studentList = new LinkedList<>();
```

Class ichida Student , Scanner LinkedList classlaridan obyekt olingan

```
while (true) {
```

```
    MENYU();
```

```
    int num = scanner.nextInt();
```

```
    switch (num) {
```

```
        case 1:
```

```
            creataStudent();
```

```
            break;
```

```
        case 2:
```

```
            getById();
```

```
            break;
```

```
        case 3:
```

```

        deleteStudent();

        break;

    case 4:

        update();

        break;

    case 5:

        allStudent();

        break;

    default:

        throw new IllegalStateException("Unexpected value: " + num);} }

```

Ushbu code dasturiy taminotni doyimiligini taminlab beradi yani dasturni whilega ga orab cheksiz sikiligini taminlab beradi

```

private void MENYU() {

    System.out.println("MENU");

    System.out.println("1:Create Student");
}

```



```
System.out.println("2: get By ID Student");

System.out.println("3: Delete Student");

System.out.println("4: Update");

System.out.println("5: All Student");

System.out.print(" enter number:");

}
```

Ushbu Code Dasturiy taminotni Menusi bu metod foydalanuvchiga mumkun bolgan ammlar.

```
private void allStudent() {

    studentList.forEach(System.out::println);

}
```

Ushbu code allStudent metodi Studentlarni royxatini korsatadi.

```
private void deleteStudent() {  
  
    System.out.print("Studentid:");  
  
    int id = scanner.nextInt();  
  
    for (Student b : studentList) {  
  
        if (b.getUserID().equals(id)) {  
  
            studentList.remove(b);  
  
        }  
  
    }  
  
    System.out.println("Student ocirildi");  
  
}
```

Ushbu code deleteStudent metodi Studenti ochirish uchun.

```
private void getByld() {  
  
    System.out.println("Studentid");  
  
    int id = scanner.nextInt();  
  
    for (Student b : studentList) {  
  
        if (b.getId().equals(id)) {  
  
            System.out.println(b);  
  
        }  
  
    }  
  
}
```

Ushbu code getByld metodi Studentni Id si boyicha topadi.

```
PRIVATE VOID UPDATE() {  
  
    SYSTEM.OUT.PRINTLN("STUDENT_ID:");  
  
    INT ID = SCANNER.NEXTINT();  
  
    FOR (STUDENT S : STUDENTLIST) {  
  
        IF (S.GETUSERID().EQUALS(ID)) {  
  
            SYSTEM.OUT.PRINT("ENTER NEW NAME:");  
  
            S.SETNAME(SCANNER.NEXT());  
  
            SYSTEM.OUT.PRINTLN("ENTER NEW SURNAME:");  
  
            S.SETSURNAME(SCANNER.NEXT());  
  
            SYSTEM.OUT.PRINTLN("USER NAME VA SURNAME OZGARTIRILDI");  
  
        }  
  
    }  
  
}
```

Ushbu code Update metodi Student malumotlarini ozgartiradi(name,surname)

```
private void createStudent() {  
  
    System.out.print("id:");  
  
    Integer id = scanner.nextInt();  
  
    System.out.println("UserID");  
  
    Integer userID = scanner.nextInt();  
  
    System.out.print("name:");  
  
    String name = scanner.next();  
  
    System.out.print("surname:");  
  
    String surname = scanner.next();  
  
    System.out.print("GroupName:");  
  
    String groupName = scanner.next();  
  
    System.out.print("Ball:");  
  
    Integer ball = scanner.nextInt();  
  
    addStudent(id, userID, name, surname, groupName, ball); }  
  

```

Ushbu code createStudent metodi Studenti yaratadi va addStudent metodiga yuboradi.

```
private void addStudent(Integer id, Integer userID, String name, String  
surname, String groupName, Integer ball) {  
  
    Student student1 = new Student();  
  
    student1.setId(id);  
  
    student1.setUserID(userID);  
  
    student1.setName(name);  
  
    student1.setSurname(surname);  
  
    student1.setGroupName(groupName);  
  
    student1.setBall(ball);  
  
    studentList.add(student1);  
  
}
```

Ushbu code addStudent metodi Studentlani Listga qoshadi.

```
public void addStudents() {  
  
    addStudent(1, 1, "aziz", "avazov", "shunqorla", 15);  
  
    addStudent(1, 2, "azim", "raximov", "shunqorla", 15);  
  
    addStudent(1, 3, "rustam", "shunqorla", "shunqorla", 15);  
  
    addStudent(1, 4, "xusan", "sayidov", "shunqorla", 15);  
  
    addStudent(1, 5, "xasan", "asadov", "shunqorla", 15);  
  
    addStudent(2, 6, "jamshit", "xasanov", "TITAN", 10);  
  
    addStudent(2, 7, "azimsher", "avazov", "TITAN", 10);  
  
    addStudent(2, 8, "moxir", "ibragimov", "TITAN", 10);  
  
    addStudent(2, 9, "sobit", "avazov", "TITAN", 10);  
  
    addStudent(2, 10, "najmidin", "tolipov", "TITAN", 10);  
  
    addStudent(3, 11, "abdulaziz", "mominov", "GOLD", 30);  
  
    addStudent(3, 12, "shoxrux", "azmib boyev", "GOLD", 30);  
  
    addStudent(3, 13, "Abduqodir", "Yolchiyev", "GOLD", 30);  
}
```

```
addStudent(3, 14, "jamolidin", "avazov", "GOLD", 30);  
  
addStudent(3, 15, "johongir", "obidxonov", "GOLD", 30);  
  
addStudent(4, 16, "Nematila", "nigmatov", "G2", 25);  
  
addStudent(4, 17, "nosirbe", "sariboyev", "G2", 25);  
  
addStudent(4, 18, "shukur", "mamotqulov", "G2", 25);  
  
addStudent(4, 19, "dilimurod", "shukurov", "G2", 25);  
  
addStudent(4, 20, "rasul", "xamdamov", "G2", 25);  
  
}
```

Ushbu code Student lar royhati.


```
public class main {  
  
    public static void main(String[] args) {  
  
        CRUDStudent crudStudent=new CRUDStudent();  
  
        crudStudent.start();  
  
    }  
  
}
```

Ushbu Class da CRUDStudent Classiga extension olingan va class ichidagi start metodi
ishga tushurilgan

XULOSA

Xulosa qilib aytadgan bo'lsak tuzilgan programma studentlar id isini va ism familyalarini saqlaydi va ballarini xisoblab beradi

So'zlar soni: 1704