Câu 1: Xây dựmg một chương trình nhập vào 2 số tự nhiên a và b. Tính tổng các số x thỏa mạn cả các điều kiện sau: a<=x <= b; x chia hết cho 4 nhưng không chia hết cho 100.

import java.util.Scanner;

public class main{

public static void main(String[] args){

Scanner scanf = new Scanner(System.in);

System.out.print("Nhap a: ");

int a = scanf.nextInt();

System.out.print("Nhap b: ");

int b = scanf.nextInt();

if(a>b){

int tmp = a;

a =b;

b=tmp;

}

int sum =0;

for(int x = a;x<=b;x++){

if(x%4 == 0 && x%100 != 0) sum +=x;

}

System.out.println("Tong cac so thoa man:" + sum);

}

}

Câu 2: Xây dựng 1 lớp theo yêu cầu sau:

|  |
| --- |
| VideoFile |
| private int bitrate;  private int length;  private String name; |
| public VideoFile(String); // default bitrate = 4194304  public VideoFile(String, int); // default length = 0;  public VideoFile(String, int, int);  public int getBitrate();  public int getLength();  public void setLength(int);  public int size(); |

public class videoFile{

private int bitrate;

private int length;

private String name;

public videoFile(String name){

this.name = name;

this.bitrate = 20;

this.length = 0;

}

public videoFile(String name, int length){

this.name = name;

this.bitrate = bitrate;

this.length = 0;

}

public videoFile(String name, int length, int bitrate){

this.name = name;

this.length = length;

this.bitrate = bitrate;

}

public int setBitrate(){

return bitrate;

}

public int getLength(){

return length;

}

public void setLength(int length){

this.length = length;

}

public int size(){

return bitrate\*length;

}

}

Câu 3: Có một cửa hàng chuyên bán thiết bị công nghệ gồm có 2 nhóm sản phẩm là phần mềm và thiết bị phần cứng. Cả 2 nhóm sản phẩm này đều có điểm chung là: giá sản phẩm, mã sản phẩm, tên sản phẩm, mô tả. Riêng sản phẩm phần cứng có thêm thông tin: kích thước, khỏi lượng. Còn sản phẩm phần mềm có thông số: hỗ trợ các hệ điều hành, yêu cầu về phần cứng tối thiểu, thông số khuyến cáo, loại hình (thuê bao, hoặc mua vĩnh viên). Xây dựng chương trình quản lý danh sách các phần mềm cho cửa hàng này gồm các chức năng sau:

﻿﻿﻿a) Thêm một sản phẩm mới. ﻿﻿﻿

b) Xóa một sản phẩm khỏi danh sách dựa theo mã sản phẩm.

﻿﻿﻿c) In ra danh sách các sản phẩm. ﻿﻿﻿

d) tìm kiếm sản phẩm theo tên sản phẩm.

import java.util.Scanner;

import java.util.ArrayList;

class SanPham{

String ma;

String ten;

double gia;

String MoTa;

public SanPham(String ma, String ten, double gia, String MoTa){

this.ma = ma;

this.ten = ten;

this.gia = gia;

this.MoTa = MoTa;

}

public String getMa(){

return ma;

}

public String getTen(){

return ten;

}

public String toString(){

return "MaSP: "+ma+", TenSP: "+ten+", Gia: "+gia+", Mo ta: "+MoTa;

}

}

class PhanCung extends SanPham{

String KichThuoc;

String KhoiLuong;

public PhanCung(String ma, String ten, double gia, String MoTa, String KichThuoc, String KhoiLuong){

super(ma, ten, gia, MoTa);

this.KichThuoc = KichThuoc;

this.KhoiLuong = KhoiLuong;

}

public String toString(){

return super.toString() + ", Kich thuoc: "+KichThuoc+", Khoi luong: "+KhoiLuong;

}

}

class PhanMem extends SanPham{

String HoTro;

String YeuCau;

String ThongSo;

String LoaiHinh;

public PhanMem(String ma, String ten, double gia, String MoTa, String HoTro, String YeuCau, String ThongSo, String LoaiHinh){

super(ma, ten, gia, MoTa);

this.HoTro = HoTro;

this.YeuCau = YeuCau;

this.ThongSo = ThongSo;

this.LoaiHinh = LoaiHinh;

}

public String toString(){

return super.toString() + ", Ho tro cac he dieu hanh: "+HoTro+", Yeu cau ve phan cung toi thieu: "+YeuCau+", Thong so khuyen cao: "+ThongSo+", Loai hinh(thue bao, hoac mua vinh vien): "+LoaiHinh;

}

}

class QLSanPham{

private ArrayList<SanPham> ds = new ArrayList<>();

public void them(SanPham sanpham){

ds.add(sanpham);

}

public void xoa(String ma){

for(SanPham sanpham : ds){

if (sanpham.getMa().equals(ma)){

ds.remove(sanpham);

return;

}

}

System.out.println("Khong tim thay san pham voi ma: "+ma);

}

public void in(){

if(ds.isEmpty()) System.out.println("Danh sach san pham trong.");

else{

for(SanPham sanpham : ds) System.out.println(sanpham);

}

}

public void TimKiem(String ten){

boolean found = false;

for(SanPham sanpham : ds){

if(sanpham.getTen().equalsIgnoreCase(ten)){

System.out.println(sanpham);

found = true;

}

}

if(!found) System.out.println("Khong tim thay san pham voi ten tren: ");

}

}

public class QLCuaHang{

public static void main(String[] args){

QLSanPham ql = new QLSanPham();

Scanner scanf = new Scanner(System.in);

int chon;

do{

System.out.println("-----MENU------");

System.out.println("1. Them san pham phan mem moi.");

System.out.println("2. Them san pham phan cung moi.");

System.out.println("3. Xoa san pham theo ma.");

System.out.println("4. In danh sach san pham.");

System.out.println("5. Tim kiem san pham theo ten.");

System.out.println("6. Thoat.");

System.out.print("Chon chuc nang: ");

chon = scanf.nextInt();

scanf.nextLine();

switch(chon){

case 1:

System.out.print("Nhap ma san pham: ");

String maPM = scanf.nextLine();

System.out.print("Nhap ten san pham: ");

String tenPM = scanf.nextLine();

System.out.print("Nhap gia san pham: ");

Double giaPM = scanf.nextDouble();

scanf.nextLine();

System.out.print("Nhap mo ta san pham: ");

String MoTaPM = scanf.nextLine();

System.out.print("Nhap ho tro he dieu hanh: ");

String HoTro = scanf.nextLine();

System.out.print("Nhap yeu cap phan cung toi thieu: ");

String YeuCau = scanf.nextLine();

System.out.print("Nhap thong so khuyen cao: ");

String ThongSo = scanf.nextLine();

System.out.print("Nhap loai hinh( thue bao, hoac mua vinh vien): ");

String LoaiHinh = scanf.nextLine();

PhanMem phanmem = new PhanMem(maPM, tenPM, giaPM, MoTaPM, HoTro, YeuCau, ThongSo, LoaiHinh);

ql.them(phanmem);

break;

case 2:

System.out.print("Nhap ma san pham: ");

String maPC = scanf.nextLine();

System.out.print("Nhap ten san pham: ");

String tenPC = scanf.nextLine();

System.out.print("Nhap gia san pham: ");

Double giaPC = scanf.nextDouble();

scanf.nextLine();

System.out.print("Nhap mo ta san pham: ");

String MoTaPC = scanf.nextLine();

System.out.print("Nhap kich thuoc san pham: ");

String KichThuoc = scanf.nextLine();

System.out.print("Nhap khoi luong san pham: ");

String KhoiLuong = scanf.nextLine();

PhanCung phancung = new PhanCung(maPC, tenPC, giaPC, MoTaPC, KichThuoc, KhoiLuong);

ql.them(phancung);

break;

case 3:

System.out.print("Nhap ma san pham can xoa: ");

String maCanXoa = scanf.nextLine();

ql.xoa(maCanXoa);

break;

case 4:

ql.in();

break;

case 5:

System.out.print("Nhap ten san pham can tim: ");

String tenCanTim = scanf.nextLine();

ql.TimKiem(tenCanTim);

break;

case 6:

System.out.print("Thoat chuong trinh");

break;

default:

System.out.print("Lua chon khong hop le. Vui long chon lai.");

}

}

while(chon != 6);

scanf.close();

}

}

Trường hợp không cho sử dụng thư viện ArraysList

import java.util.Arrays;

import java.util.Comparator;

import java.util.Scanner;

// Lớp cha Sản phẩm

class SanPham {

String maSanPham;

String tenSanPham;

double giaSanPham;

String moTa;

public SanPham(String maSanPham, String tenSanPham, double giaSanPham, String moTa) {

this.maSanPham = maSanPham;

this.tenSanPham = tenSanPham;

this.giaSanPham = giaSanPham;

this.moTa = moTa;

}

public String getMaSanPham() {

return maSanPham;

}

public String getTenSanPham() {

return tenSanPham;

}

@Override

public String toString() {

return "MaSP: " + maSanPham + ", TenSP: " + tenSanPham + ", Gia: " + giaSanPham + ", Mo ta: " + moTa;

}

}

// Lớp con Sản phẩm phần mềm

class PhanMem extends SanPham {

String hoTroHeDieuHanh;

String yeuCauPhanCungToiThieu;

String thongSoKhuyenCao;

String loaiHinh;

public PhanMem(String maSanPham, String tenSanPham, double giaSanPham, String moTa,

String hoTroHeDieuHanh, String yeuCauPhanCungToiThieu, String thongSoKhuyenCao, String loaiHinh) {

super(maSanPham, tenSanPham, giaSanPham, moTa);

this.hoTroHeDieuHanh = hoTroHeDieuHanh;

this.yeuCauPhanCungToiThieu = yeuCauPhanCungToiThieu;

this.thongSoKhuyenCao = thongSoKhuyenCao;

this.loaiHinh = loaiHinh;

}

@Override

public String toString() {

return super.toString() + ", He dieu hanh: " + hoTroHeDieuHanh + ", Yeu cau phan cung toi thieu: " + yeuCauPhanCungToiThieu +

", Thong so khuyen cao: " + thongSoKhuyenCao + ", Loai hinh: " + loaiHinh;

}

}

// Lớp con Sản phẩm phần cứng

class PhanCung extends SanPham {

String kichThuoc;

String khoiLuong;

public PhanCung(String maSanPham, String tenSanPham, double giaSanPham, String moTa, String kichThuoc, String khoiLuong) {

super(maSanPham, tenSanPham, giaSanPham, moTa);

this.kichThuoc = kichThuoc;

this.khoiLuong = khoiLuong;

}

@Override

public String toString() {

return super.toString() + ", Kich thuoc: " + kichThuoc + ", Khoi luong: " + khoiLuong;

}

}

// Lớp quản lý danh sách sản phẩm

class QuanLySanPham {

private SanPham[] danhSachSanPham = new SanPham[0];

// Thêm một sản phẩm mới

public void themSanPham(SanPham sanPham) {

danhSachSanPham = Arrays.copyOf(danhSachSanPham, danhSachSanPham.length + 1);

danhSachSanPham[danhSachSanPham.length - 1] = sanPham;

}

// Xóa một sản phẩm theo mã sản phẩm

public void xoaSanPham(String maSanPham) {

int index = -1;

for (int i = 0; i < danhSachSanPham.length; i++) {

if (danhSachSanPham[i].getMaSanPham().equals(maSanPham)) {

index = i;

break;

}

}

if (index != -1) {

SanPham[] temp = new SanPham[danhSachSanPham.length - 1];

System.arraycopy(danhSachSanPham, 0, temp, 0, index);

System.arraycopy(danhSachSanPham, index + 1, temp, index, danhSachSanPham.length - index - 1);

danhSachSanPham = temp;

} else {

System.out.println("Không tìm thấy sản phẩm với mã: " + maSanPham);

}

}

// In danh sách các sản phẩm

public void inDanhSachSanPham() {

if (danhSachSanPham.length == 0) {

System.out.println("Danh sách sản phẩm trống.");

} else {

for (SanPham sanPham : danhSachSanPham) {

System.out.println(sanPham);

}

}

}

// Tìm kiếm sản phẩm theo tên

public void timKiemSanPham(String tenSanPham) {

boolean found = false;

for (SanPham sanPham : danhSachSanPham) {

if (sanPham.getTenSanPham().equalsIgnoreCase(tenSanPham)) {

System.out.println(sanPham);

found = true;

}

}

if (!found) {

System.out.println("Không tìm thấy sản phẩm với tên: " + tenSanPham);

}

}

}

public class QuanLyCuaHang {

public static void main(String[] args) {

QuanLySanPham quanLySanPham = new QuanLySanPham();

Scanner scanner = new Scanner(System.in);

int choice;

do {

System.out.println("----- MENU -----");

System.out.println("1. Thêm sản phẩm phần mềm mới");

System.out.println("2. Thêm sản phẩm phần cứng mới");

System.out.println("3. Xóa sản phẩm theo mã");

System.out.println("4. In danh sách sản phẩm");

System.out.println("5. Tìm kiếm sản phẩm theo tên");

System.out.println("6. Thoát");

System.out.print("Chọn chức năng: ");

choice = scanner.nextInt();

scanner.nextLine(); // Consume newline

switch (choice) {

case 1:

System.out.print("Nhập mã sản phẩm: ");

String maPhanMem = scanner.nextLine();

System.out.print("Nhập tên sản phẩm: ");

String tenPhanMem = scanner.nextLine();

System.out.print("Nhập giá sản phẩm: ");

double giaPhanMem = scanner.nextDouble();

scanner.nextLine(); // Consume newline

System.out.print("Nhập mô tả sản phẩm: ");

String moTaPhanMem = scanner.nextLine();

System.out.print("Nhập hỗ trợ hệ điều hành: ");

String hoTroHeDieuHanh = scanner.nextLine();

System.out.print("Nhập yêu cầu phần cứng tối thiểu: ");

String yeuCauPhanCungToiThieu = scanner.nextLine();

System.out.print("Nhập thông số khuyến cáo: ");

String thongSoKhuyenCao = scanner.nextLine();

System.out.print("Nhập loại hình (thuê bao/mua vĩnh viễn): ");

String loaiHinh = scanner.nextLine();

PhanMem phanMem = new PhanMem(maPhanMem, tenPhanMem, giaPhanMem, moTaPhanMem, hoTroHeDieuHanh,

yeuCauPhanCungToiThieu, thongSoKhuyenCao, loaiHinh);

quanLySanPham.themSanPham(phanMem);

break;

case 2:

System.out.print("Nhập mã sản phẩm: ");

String maPhanCung = scanner.nextLine();

System.out.print("Nhập tên sản phẩm: ");

String tenPhanCung = scanner.nextLine();

System.out.print("Nhập giá sản phẩm: ");

double giaPhanCung = scanner.nextDouble();

scanner.nextLine(); // Consume newline

System.out.print("Nhập mô tả sản phẩm: ");

String moTaPhanCung = scanner.nextLine();

System.out.print("Nhập kích thước sản phẩm: ");

String kichThuoc = scanner.nextLine();

System.out.print("Nhập khối lượng sản phẩm: ");

String khoiLuong = scanner.nextLine();

PhanCung phanCung = new PhanCung(maPhanCung, tenPhanCung, giaPhanCung, moTaPhanCung, kichThuoc, khoiLuong);

quanLySanPham.themSanPham(phanCung);

break;

case 3:

System.out.print("Nhập mã sản phẩm cần xóa: ");

String maCanXoa = scanner.nextLine();

quanLySanPham.xoaSanPham(maCanXoa);

break;

case 4:

quanLySanPham.inDanhSachSanPham();

break;

case 5:

System.out.print("Nhập tên sản phẩm cần tìm: ");

String tenCanTim = scanner.nextLine();

quanLySanPham.timKiemSanPham(tenCanTim);

break;

case 6:

System.out.println("Thoát chương trình.");

break;

default:

System.out.println("Lựa chọn không hợp lệ. Vui lòng chọn lại.");

}

} while (choice != 6);

scanner.close();

}

}

Câu 4: Cho một tệp tin chứa danh sách các đường dẫn truy cập vào website (URL) có dạng như sau (các dạng được kiểm tra theo thử tự ưu tiên từ trên xuống dưới):

OPTIONS /any/path

PUT/product//upload/:sha256

GET /product//files/:sha256

GET/product/:id

GET /product/product-slug

Trong đó :id là chỉ số nhận dạng sản phẩm là một số gồm 12 byte biểu diễn dưới đạng thập lục, product-siug chỉ gồm các ký tự chữ cái, chữ số và dầu gạch ngang. Thực hiện xử lý các thông tin trong file và in ra màn hình cũng như ghi ra một file khác các dữ liệu dạng như sau.

CORS: /any/path

Upload: ( productid: ":id", filename: ":sha256" }

Read: (productid: ":id", filename: ":sha256" }

Render: (productid: ":id" }

FindProductid: ( path: "product-slug" }

import java.io.\*;

import java.util.regex.\*;

public class XuLyURL {

public static void main(String[] args) {

String DauVao = "input.txt";

String DauRa = "output.txt";

try (BufferedReader br = new BufferedReader(new FileReader(DauVao));

BufferedWriter bw = new BufferedWriter(new FileWriter(DauRa))) {

String dong;

while ((dong = br.readLine()) != null) {

String ketQua = xuLyURL(dong);

if (ketQua != null) {

System.out.println(ketQua);

bw.write(ketQua);

bw.newLine();

}

}

} catch (IOException e) {

e.printStackTrace();

}

}

private static String xuLyURL(String url) {

Pattern mauOPTIONS = Pattern.compile("OPTIONS\\s+/any/path");

Pattern mauPUT = Pattern.compile("PUT\\s+/product/([a-fA-F0-9]{24})/upload/([a-fA-F0-9]{64})");

Pattern mauGETFile = Pattern.compile("GET\\s+/product/([a-fA-F0-9]{24})/files/([a-fA-F0-9]{64})");

Pattern mauGETID = Pattern.compile("GET\\s+/product/([a-fA-F0-9]{24})");

Pattern mauGETSlug = Pattern.compile("GET\\s+/product/([a-zA-Z0-9\\-]+)");

Matcher khop;

if ((khop = mauOPTIONS.matcher(url)).matches()) {

return "CORS: /any/path";

} else if ((khop = mauPUT.matcher(url)).matches()) {

return String.format("Upload: { productid: \"%s\", filename: \"%s\" }", khop.group(1), khop.group(2));

} else if ((khop = mauGETFile.matcher(url)).matches()) {

return String.format("Read: { productid: \"%s\", filename: \"%s\" }", khop.group(1), khop.group(2));

} else if ((khop = mauGETID.matcher(url)).matches()) {

return String.format("Render: { productid: \"%s\" }", khop.group(1));

} else if ((khop = mauGETSlug.matcher(url)).matches()) {

return String.format("FindProductid: { path: \"%s\" }", khop.group(1));

}

return null;

}

}