

Bài 1: Tính chu vi và diện tích các hình

Viết chương trình tính chu vi và diện tích của một số hình như sau:

- Hình tròn
- Hình chữ nhật
- Hình tam giác

Code:

*/*Hình tròn*/*

import math

class Hình_Tròn():

```
def __init__(self, r):  
    self.r = r
```

```
def tinh_chu_vi(self):  
    return 2*math.pi*self.r
```

```
def tinh_dien_tich(self):  
    return math.pi*math.pow(self.r,2)
```

```
if __name__ == '__main__':  
    Htron = Hình_Tròn(1)  
    print("Ban kinh hình tròn: ", Htron.r)  
    print("Chu vi hình tròn: ", Htron.tinh_chu_vi())  
    print("Diện tích hình tròn: ", Htron.tinh_dien_tich())
```

*/*Hình chữ nhật*/*

class Hình_Chữ_Nhật:

```
def __init__(self, chieudai, chieurong):  
    self.chieudai = chieudai  
    self.chieurong = chieurong
```

```
def tinh_chu_vi(self):  
    return (self.chieudai + self.chieurong)*2
```

```
def tinh_dien_tich(self):  
    return (self.chieudai* self.chieurong)
```

```

if __name__ == '__main__':
    HChuNhat = Hinh_Chua_Nhat(2,3)
    print("Chieu dai hinh chu nhat: ", HChuNhat.chieudai)
    print("Chieu rong hinh chu nhat: ", HChuNhat.chieurong)
    print("Chu vi hinh tron: ", HChuNhat.tinh_chu_vi())
    print("Dien tich hinh tron: ", HChuNhat.tinh_dien_tich())
/*Hình tam giác*/

```

```

import math

```

```

class Hinh_Tam_Giac:

```

```

    def __init__(self, a, b, c):
        self.a = a
        self.b = b
        self.c = c

```

```

    def tinh_chu_vi(self):
        return self.a + self.b + self.c

```

```

    def tinh_dien_tich(self):
        # Su dung cong thuc heron
        p = (self.a + self.b + self.c)/2
        s = math.sqrt(p*(p - self.a)*(p - self.b)*(p - self.c))
        return s

```

```

if __name__ == '__main__':
    HTamGiac = Hinh_Tam_Giac(2,3,4)
    print("Ba canh cua tam giac: ", HTamGiac.a, HTamGiac.b, HTamGiac.c)
    print("Chu vi tam giac: ", HTamGiac.tinh_chu_vi())
    print("Dien tich tam giac: ", HTamGiac.tinh_dien_tich())

```

Bài 2: Giải phương trình bậc nhất

```

class PhuongTrinhBacNhat():
    def __init__(self, a, b):

    def TimNghiem(self):

if __name__ == '__main__':

```

Bài 3: Giải phương trình bậc hai

```

class PhuongTrinhBacHai():
    def __init__(self, _a, _b, _c):
        self.a=_a
        self.b=_b
        self.c=_c
    def TimNghiem(self):
        nghiem=""
        if self.a==0:
            nghiem='Phuong trinh khong hop le'
        else:
            delta=self.b*self.b-4*self.a*self.c
            if delta<0:
                nghiem='Phuong trinh vo nghiem'
            elif delta==0:
                nghiem='Phuong trinh co nghiem kep x1 = x2 = '+ str(((self.b)/(2*self.a)))
            else:
                x1=(-self.b+math.sqrt(delta))/(2*self.a)
                x2=(-self.b-math.sqrt(delta))/(2*self.a)
                nghiem='Phuong trinh co hai nghiem phan biet \n'
                nghiem+='x1 = ' + str(x1)
                nghiem+='x2 = ' + str(x2)
            return nghiem
if __name__ == '__main__':
    PT = PhuongTrinhBacHai(1,-2,1)
    print("Phuong trinh bac hai co cac he so: ", PT.a, PT.b, PT.c)
    Nghiem=PT.TimNghiem()
    print(Nghiem)

```

Bài 4: Viết chương trình quản lý các CD(Công Đoàn) như sau:

Thông tin CD:

Nhập tên CD: Happy New Year

Nhập tên ca sỹ: ABBA

Nhập số bài hát: 7

Nhập giá thành: 185000

--- Danh sách CD: ---

Happy New Year - ABBA - 7 - 185000

Tổng giá thành: 185000

Tiếp tục nhập: 1: Có, 0: Không 1

Nhập tên CD: Chat với Mozart

Nhập tên ca sỹ: Mỹ Linh

Nhập số bài hát: 8

Nhập giá thành: 245000

--- Danh sách CD: ---

Happy New Year - ABBA - 7 - 185000

Chat với Mozart - Mỹ Linh - 8 - 245000

Tổng giá thành: 430000

Tiếp tục nhập: 1: Có, 0: Không 0

Người dùng lần lượt nhập thông tin của các CD, chương trình sẽ quản lý thông tin các CD này và in ra danh sách các CD kèm theo tổng số tiền của các CD

Bài 4: Bài tập quản lý sinh viên trong Python

Đề bài: Viết chương trình quản lý sinh viên trong Python. Mỗi đối tượng sinh viên có các thuộc tính sau: id, tên, giới tính, tuổi, điểm toán, điểm lý, điểm hóa, điểm trung bình và học lực.

Id là mã sinh viên tự động tăng.

Điểm trung bình là giá trị trung bình của 3 môn toán, lý và hóa.

Học lực được tính như sau:

- Giỏi: nếu điểm trung bình lớn hơn hoặc bằng 8.
- Khá: nếu điểm trung bình nhỏ hơn 8 và lớn hơn hoặc bằng 6.5.
- Trung Bình: nếu điểm trung bình nhỏ hơn 6.5 và lớn hơn hoặc bằng 5.
- Yếu: nếu điểm trung bình nhỏ hơn 5.

Yêu cầu: tạo ra một menu với các chức năng sau:

1. Thêm sinh viên.
2. Cập nhật thông tin sinh viên bởi ID.
3. Xóa sinh viên bởi ID.
4. Tìm kiếm sinh viên theo tên.

- 5. Sắp xếp sinh viên theo điểm trung bình (GPA).

Class:

class SinhVien:

```
def __init__(self, id, name, sex, age, diemToan, diemLy, diemHoa):
    self._id = id
    self._name = name
    self._sex = sex
    self._age = age
    self._diemToan = diemToan
    self._diemLy = diemLy
    self._diemHoa = diemHoa
    self._diemTB = 0
    self._hocLuc = ""
```

Tạo class quản lý sinh viên

listSinhVien = []

Sau đây là danh sách các phương thức mà chúng ta sẽ sử dụng trong chương trình quản lý sinh viên:

```
def generateID(self)
def soLuongSinhVien(self)
def nhapSinhVien(self)
def updateSinhVien(self, ID)
def sortByID(self)
def sortByName(self)
def sortByDiemTB(self)
def findByID(self, ID)
def findByName(self, keyword)
def deleteById(self, ID)
def tinhDTB(self, sv:SinhVien)
def showSinhVien(self, listSV)
def getListSinhVien(self)
```

Hàm tạo ID tăng dần cho nhân viên

```
def generateID(self):
    maxId = 1
    if (self.soLuongSinhVien() > 0):
        maxId = self.listSinhVien[0]._id
    for sv in self.listSinhVien:
        if (maxId < sv._id):
            maxId = sv._id
```

```
    maxId = maxId + 1
return maxId
```

```
def soLuongSinhVien(self):
    return self.listSinhVien.__len__()
```

```
def nhapSinhVien(self):
    # Khởi tạo một sinh viên mới
    svId = self.generateID()
    name = input("Nhập ten sinh vien: ")
    sex = input("Nhập giới tính sinh vien: ")
    age = int(input("Nhập tuổi sinh vien: "))
    diemToan = float(input("Nhập diem toan: "))
    diemLy = float(input("Nhập diem Ly: "))
    diemHoa = float(input("Nhập diem Hoa: "))
    sv = SinhVien(svId, name, sex, age, diemToan, diemLy, diemHoa)
    self.tinhDTB(sv)
    self.xepLoaiHocLuc(sv)
    self.listSinhVien.append(sv)
```

```
def updateSinhVien(self, ID):
    # Tìm kiếm sinh viên trong danh sách listSinhVien
    sv:SinhVien = self.findByID(ID)
    # Nếu sinh viên tồn tại thì cập nhật thông tin sinh viên
    if (sv != None):
        # nhập thông tin sinh viên
        name = input("Nhập ten sinh vien: ")
        sex = input("Nhập giới tính sinh vien: ")
        age = int(input("Nhập tuổi sinh vien: "))
        diemToan = float(input("Nhập diem toan: "))
        diemLy = float(input("Nhập diem Ly: "))
        diemHoa = float(input("Nhập diem Hoa: "))
        # cập nhật thông tin sinh viên
        sv._name = name
        sv._sex = sex
        sv._age = age
        sv._diemToan = diemToan
        sv._diemLy = diemLy
        sv._diemHoa = diemHoa
        self.tinhDTB(sv)
        self.xepLoaiHocLuc(sv)
    else:
        print("Sinh vien co ID = { } khong ton tai.".format(ID))
```

```
# Hàm xóa sinh viên theo ID
```

```

def deleteById(self, ID):
    isDeleted = False
    # tìm kiếm sinh viên theo ID
    sv = self.findByID(ID)
    if (sv != None):
        self.listSinhVien.remove(sv)
        isDeleted = True
    return isDeleted

# Hàm tìm kiếm sinh viên theo tên
# Trả về một danh sách sinh viên
def findByName(self, keyword):
    listSV = []
    if (self.soLuongSinhVien() > 0):
        for sv in self.listSinhVien:
            if (keyword.upper() in sv._name.upper()):
                listSV.append(sv)
    return listSV

def tinhDTB(self, sv:SinhVien):
    diemTB = (sv._diemToan + sv._diemLy + sv._diemHoa) / 3
    # làm tròn điểm trung bình với 2 chữ số thập phân
    sv._diemTB = math.ceil(diemTB * 100) / 100

#Hàm xếp loại học lực cho nhân viên
def xepLoaiHocLuc(self, sv:SinhVien):
    if (sv._diemTB >= 8):
        sv._hocLuc = "Gioi"
    elif (sv._diemTB >= 6.5):
        sv._hocLuc = "Kha"
    elif (sv._diemTB >= 5):
        sv._hocLuc = "Trung Binh"
    else:
        sv._hocLuc = "Yeu"

def sortByDiemTB(self):
    self.listSinhVien.sort(key=lambda x: x._diemTB, reverse=False)

#Hàm sắp xếp danh sách sinh viên theo tên tăng dần
def sortByName(self):
    self.listSinhVien.sort(key=lambda x: x._name, reverse=False)

# Hàm hiển thị danh sách sinh viên ra màn hình console
def showSinhVien(self, listSV):
    # hiển thị tiêu đề cột
    print("{:<8} {:<18} {:<8} {:<8}{:<8} {:<8} {:<8} {:<8} {:<8}"

```

```

        .format("ID", "Name", "Sex", "Age", "Toan", "Ly", "Hoa", "Diem TB", "Hoc
Luc"))
    # hien thi danh sach sinh vien
    if (listSV.__len__() > 0):
        for sv in listSV:
            print("{:<8} {:<18} {:<8} {:<8}{:<8} {:<8} {:<8} {:<8} {:<8}")

```

Class Main.py:

```

qlsv = QuanLySinhVien()
while (1==1):
    print("\nCHUONG TRINH QUAN LY SINH VIEN C#")
    print("*****MENU*****")
    print("*** 1. Them sinh vien.                ***")
    print("*** 2. Cap nhat thong tin sinh vien boi ID.        ***")
    print("*** 3. Xoa sinh vien boi ID.                    ***")
    print("*** 4. Tim kiem sinh vien theo ten.                ***")
    print("*** 5. Sap xep sinh vien theo diem trung binh (GPA). ***")
    print("*** 6. Sap xep sinh vien theo ten.                ***")
    print("*** 7. Sap xep sinh vien theo ID.                ***")
    print("*** 8. Hien thi danh sach sinh vien.              ***")
    print("*** 0. Thoat                                     ***")
    print("*****")

    key = int(input("Nhap tuy chon: "))
    if (key == 1):
        print("\n1. Them sinh vien.")
        qlsv.nhapSinhVien()
        print("\nThem sinh vien thanh cong!")
    elif (key == 2):
        if (qlsv.soLuongSinhVien() > 0):
            print("\n2. Cap nhat thong tin sinh vien. ")
            print("\nNhap ID: ")
            ID = int(input())
            qlsv.updateSinhVien(ID)
        else:
            print("\nSanh sach sinh vien trong!")
    elif (key == 3):
        if (qlsv.soLuongSinhVien() > 0):
            print("\n3. Xoa sinh vien.")
            print("\nNhap ID: ")
            ID = int(input())
            if (qlsv.deleteById(ID)):
                print("\nSinh vien co id = ", ID, " da bi xoa.")
            else:

```



```

        print("\nSinh vien co id = ", ID , " khong ton tai.")
    else:
        print("\nSanh sach sinh vien trong!")
elif (key == 4):
    if (qlsv.soLuongSinhVien() > 0):
        print("\n4. Tim kiem sinh vien theo ten.")
        print("\nNhap ten de tim kiem: ")
        name = input()
        searchResult = qlsv.findByName(name)
        qlsv.showSinhVien(searchResult)
    else:
        print("\nSanh sach sinh vien trong!")
elif (key == 5):
    if (qlsv.soLuongSinhVien() > 0):
        print("\n5. Sap xep sinh vien theo diem trung binh (GPA).")
        qlsv.sortByDiemTB()
        qlsv.showSinhVien(qlsv.getListSinhVien())
    else:
        print("\nSanh sach sinh vien trong!")
elif (key == 6):
    if (qlsv.soLuongSinhVien() > 0):
        print("\n6. Sap xep sinh vien theo ten.")
        qlsv.sortByName()
        qlsv.showSinhVien(qlsv.getListSinhVien())
    else:
        print("\nSanh sach sinh vien trong!")
elif (key == 7):
    if (qlsv.soLuongSinhVien() > 0):
        print("\n6. Sap xep sinh vien theo ID.")
        qlsv.sortByID()
        qlsv.showSinhVien(qlsv.getListSinhVien())
    else:
        print("\nSanh sach sinh vien trong!")
elif (key == 8):
    if (qlsv.soLuongSinhVien() > 0):
        print("\n7. Hien thi danh sach sinh vien.")
        qlsv.showSinhVien(qlsv.getListSinhVien())
    else:
        print("\nSanh sach sinh vien trong!")
elif (key == 0):
    print("\nBan da chon thoat chuong trinh!")
    break
else:
    print("\nKhong co chuc nang nay!")
    print("\nHay chon chuc nang trong hop menu.")

```

Bài 5:

Yêu cầu: Xây dựng ứng dụng quản lý danh sách các giao dịch: Mô tả: Hệ thống quản lý 2 loại giao dịch:

Giao dịch vàng: Mã giao dịch, ngày giao dịch (ngày/tháng/năm), đơn giá, số lượng, loại vàng có 3 loại 18k, 24k, 9999.

- Thành tiền được tính như sau: $\text{thành tiền} = \text{số lượng} * \text{đơn giá}$.

Giao dịch tiền tệ: Mã giao dịch, ngày giao dịch (ngày/tháng/năm), tỷ giá (cũng là đơn giá), số lượng, loại tiền tệ có 3 loại: USD, EUR, AUD, loại giao dịch mua/bán. Thành tiền được tính như sau:

- Nếu loại giao dịch là “mua” thì: $\text{thành tiền} = \text{số lượng} * \text{tỷ giá}$
- Nếu loại giao dịch là “bán” thì: $\text{thành tiền} = (\text{số lượng} * \text{tỷ giá}) * 1.05$

```
Quản lý giao dịch:
Nhập mã GD:      gd001
Nhập ngày GD:    13/03/2017
Nhập số lượng:   10
Chọn loại giao dịch: 1: Vàng, 2: Tiền Tệ:      1
Chọn loại: 18k / 24k / 9999:      18k
Nhập đơn giá:    2350000
gd001 - 13/03/2017 - 18k - 10 - 2350000 - Thành tiền = 23500000
Tổng số lượng: 10
Tổng số tiền: 23500000
Bạn muốn tiếp tục giao dịch? 1: Có, 0: Không    1
Nhập mã GD:      gd002
Nhập ngày GD:    14/03/2017
Nhập số lượng:   100
Chọn loại giao dịch: 1: Vàng, 2: Tiền Tệ:      2
Chọn loại: USD / EUR / AUD:      USD
Nhập tỷ giá:      23000
Bạn mua hay bán? 1: mua, 0: bán:              1
GD mua: gd002 - 14/03/2017 - USD - 100 - 23000 - Thành tiền = 2300000
Tổng số lượng: 100
Tổng số tiền: 2300000
Bạn muốn tiếp tục giao dịch? 1: Có, 0: Không    1
```

Dựa vào mô tả trên, hãy:

Tạo lớp GiaoDich với các thuộc tính và phương thức chung (giao dịch vàng cũng là giao dịch).

Tạo lớp GiaoDichTienTekế thừa từ lớp GiaoDich với các thuộc tính riêng và phương thức cần thiết.

Nhập xuất danh sách các giao dịch.

Tính tổng số lượng cho từng loại.

Tính tổng thành tiền cho từng loại.

