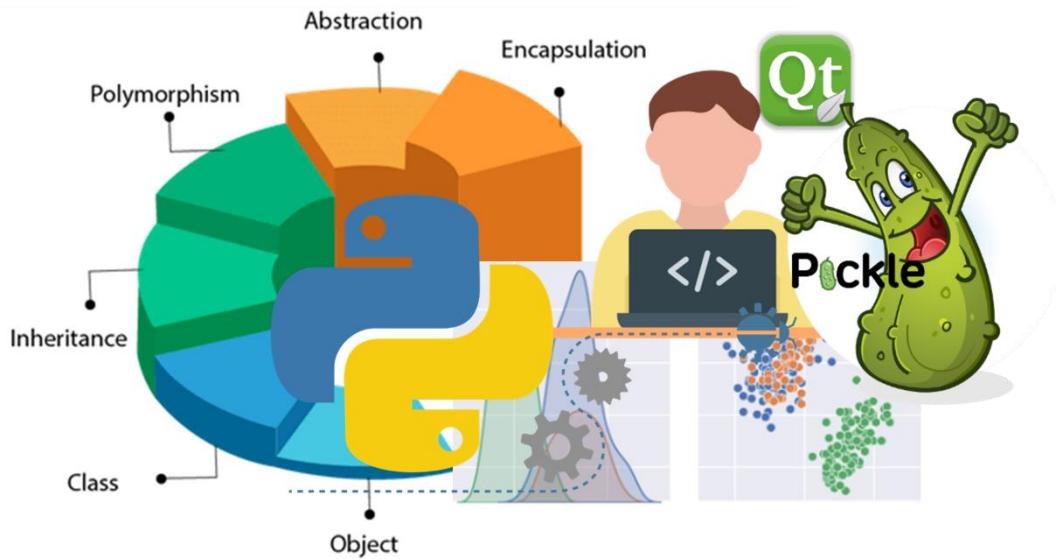


Hướng Dẫn Thực Hành

KỸ THUẬT LẬP TRÌNH



Hướng dẫn:

- Bài tập thực hành được chia làm nhiều Module
- Mỗi Module được thiết kế cho thời lượng là 3 hoặc 6 tiết thực hành tại lớp với sự hướng dẫn của Giảng viên.
- Tùy theo số tiết phân bổ, mỗi tuần học có thể thực hiện nhiều Module.
- Sinh viên phải làm tất cả các bài tập trong các Module ở tuần tương ứng. Những sinh viên chưa hoàn tất phần bài tập tại lớp có trách nhiệm tự làm tiếp tục ở nhà.
- Các bài có dấu (*) là các bài tập nâng cao dành cho sinh viên khá giỏi.

Biên soạn và tổng hợp:

NCS. ThS. Nguyễn Quang Phúc- phucnq@uel.edu.vn

ThS. Trương Hoài Phan- hoaiphan@uel.edu.vn

TS. Trần Duy Thành- thanhtd@uel.edu.vn

MỤC LỤC

MODULE 1	6
HÀM VÀ CÁC KỸ THUẬT XỬ LÝ HÀM	6
Câu 1: Trình bày cơ chế hoạt động của hàm và cho ví dụ minh họa	6
Câu 2: Phân biệt local variable và global variable	7
Câu 3: Cho biết kết quả xuất ra màn hình sau khi thực hiện mã lệnh	7
Câu 4: Cho biết kết quả xuất ra màn hình sau khi thực hiện mã lệnh	7
Câu 5: Cho biết kết quả xuất ra màn hình sau khi thực hiện mã lệnh	8
Câu 6: Trình bày cú pháp và cơ chế hoạt động của Lambda Expression	9
Câu 7: Cho biết kết quả xuất ra màn hình sau khi thực hiện mã lệnh	9
Câu 8: Viết hàm tính điểm trung bình và xếp loại.....	10
Câu 9: Đệ quy là gì? Cho biết các điều kiện để lập trình đệ quy	10
Câu 10: Trình bày và minh họa đệ quy tuyển tính	10
Câu 11: Trình bày và minh họa đệ quy đuôi	11
Câu 12: Trình bày và minh họa đệ quy nhị phân.....	11
Câu 13: Trình bày và minh họa đệ quy đa tuyển	11
Câu 14: Trình bày và minh họa đệ quy lồng	12
Câu 15: Trình bày và minh họa đệ quy tương hỗ	12
Câu 16: Viết hàm đệ quy chuyển cơ số 10 sang cơ số 2	12
Câu 17: Viết hàm đệ quy chuyển cơ số 10 sang cơ số thập lục phân.....	13
Câu 18: Viết hàm xử lý số Fibonacci (*).	14
Câu 19: Tháp Hà Nội (*).	15
Câu 20: Tạo hướng dẫn cho hàm.....	16
Câu 21: Phần mềm tính chỉnh hợp - tổ hợp.....	16
Câu 22: Viết hàm tính xác suất bóng đèn hư.....	17
Câu 23: Phần mềm bán sách mini	18
Câu 24: Phần mềm Lucky Number (*).	19
MODULE 2	21
CÁU TRÚC DỮ LIỆU TRONG PYTHON	21
Câu 25: Viết chương trình kiểm tra chuỗi đối xứng.....	21
Câu 26: Các hàm xử lý chuỗi cơ bản.....	22
Câu 27: Viết chương trình tối ưu chuỗi danh từ.....	23
Câu 28: Thống kê số trong chuỗi.....	24
Câu 29: Trích lọc số âm trong chuỗi	25
Câu 30: Tách tên bài hát và loại tập tin	25
Câu 31: Tách lấy tên các thành phần trong URL	26



Câu 32: Mô tả mảng và các thao tác trên mảng.....	26
Câu 33: Xây dựng phần mềm tương tác mảng	27
Câu 34: Các thao tác trên mảng.....	28
Câu 35: Mô tả và phân biệt cấu trúc dữ liệu tập hợp.....	28
Câu 36: Xác định kết quả khi thực thi List.....	29
Câu 37: Xác định kết quả khi thực thi List.....	29
Câu 38: Các thao tác trên List	30
Câu 39: Phần mềm thao tác trên List.....	31
Câu 40: Xử lý List đa chiều.....	33
Câu 41: Viết hàm thống kê sử dụng Tuple	33
Câu 42: Tìm các phần tử duy nhất trong một list có chứa các Tuple	34
Câu 43: Sắp xếp Sinh viên theo nhiều tiêu chí	35
Câu 44: Tính tổng các phần tử nhiều tiêu chí	35
Câu 45: Phần mềm Quản lý sách (*)	36
Câu 46: Tìm tập hợp con lớn nhất không có phần tử chung với tập hợp ban đầu.....	38
Câu 47: Kiểm tra tính đối xứng của một tập hợp các tuple	38
Câu 48: Tìm tất cả các phần tử là hợp của các tập hợp con trong một danh sách các set	39
Câu 49: Tạo dictionary từ danh sách các tuple và tính tổng các giá trị trùng lặp	39
Câu 50: Sắp xếp dictionary theo giá trị và giữ nguyên key nếu giá trị trùng nhau	40
Câu 51: Tạo dictionary chứa thông tin thống kê từ một danh sách các dictionary	41
MODULE 3	42
LẬP TRÌNH HƯỚNG ĐỐI TƯỢNG	42
Câu 52: Trình bày Khái niệm về lập trình hướng đối tượng	42
Câu 53: Phân biệt Overloading và Overriding Method.....	43
Câu 54: Giải thích chi tiết Alias và cơ chế gom rác tự động	43
Câu 55: Trình bày các kỹ thuật khai báo phương thức tĩnh và ý nghĩa	43
Câu 56: Trình bày các kỹ thuật kế thừa và ý nghĩa	44
Câu 57: Trình bày kỹ thuật Singleton và ý nghĩa sử dụng	44
Câu 58: Xây dựng lớp Nhân viên	44
Câu 59: Xây dựng lớp Phân Số	45
Câu 60: Phần mềm quản lý Sinh viên.....	46
Câu 61: Phần mềm quản lý Sinh viên.....	47
Câu 62: Phần mềm Quản lý Sản phẩm	48
Câu 63: Phần mềm quản lý Sản phẩm – kế thừa	48
Câu 64: Phần mềm quản lý Nhân viên – Kế thừa	50
Câu 65: Phần mềm quản lý Nhân viên – Kế thừa(*).....	51
Câu 66: Phần mềm quản lý Dân số (*).....	52
Câu 67: Phần mềm quản lý Đối tượng toán học	52
Câu 68: Phần mềm quản lý Bán hàng (*).....	54

Câu 69: Phần mềm quản lý thư viện (*)	55
Câu 70: Phần mềm quản lý đặt phòng khách sạn (*)	56
Câu 71: Phần mềm mô phỏng hệ thống ATM (*)	56
Câu 72: Phần mềm mô phỏng hệ thống đặt vé xe bus/tàu hỏa (*)	57
Câu 73: Phần mềm mô phỏng hệ thống quản lý bệnh viện (*)	57
MODULE 4	58
THAO TÁC VỚI TẬP TIN	58
Câu 74: Trình bày kỹ thuật xử lý tập tin .txt	58
Câu 75: Trình bày kỹ thuật xử lý tập tin XML	58
Câu 76: Trình bày kỹ thuật xử lý tập tin JSON	59
Câu 77: Trình bày kỹ thuật xử lý tập tin Excel (CSV, XLS, XLSX)	59
Câu 78: Trình bày kỹ thuật xử lý Pickle File	60
Câu 79: Xử lý số trong Text File	60
Câu 80: Chương trình quản lý Sản phẩm – Text File	62
Câu 81: Chương trình quản lý Employee – XML	66
Câu 82: Chương trình quản lý Product – XML	70
Câu 83: Chuyển đổi String JSON qua Python Object	71
Câu 84: Chuyển đổi Python Object qua String JSON	72
Câu 85: Chương trình quản lý Assets – JSON (*).	72
Câu 86: Xử lý dữ liệu trong tập tin CSV	74
Câu 87: Chương trình book Hotel – CSV (*).	75
Câu 88: Xử lý dữ liệu trong tập tin Excel XLSX	77
Câu 89: Viết phần mềm quản lý Nhân viên - Excel	79
Câu 90: Phần mềm quản lý thiết bị XML (*)	80
Câu 91: Quản lý Sản phẩm - Pickle File	81
Câu 92: Phần mềm quản lý Sinh viên – Pickle File(*).	85
Câu 93: Xử lý đa phương tiện	86
MODULE 5	87
THIẾT KẾ GIAO DIỆN TƯƠNG TÁC NGƯỜI DÙNG NÂNG CAO	87
Câu 94: Nạp dữ liệu và tương tác QCombobox	87
Câu 95: Nạp dữ liệu hướng đối tượng và tương tác QCombobox	88
Câu 96: Quản lý thông tin Nhân viên	89
Câu 97: Nạp dữ liệu và tương tác QListWidget	90
Câu 98: Nạp dữ liệu và tương tác QListWidget	91
Câu 99: Nạp dữ liệu và tương tác QListWidget	92
Câu 100: Quản lý nhân viên - QListWidget	93
Câu 101: Phần mềm Food Management - QDateEdit	95
Câu 102: Phần mềm Tracking Tasks - QDateTimeEdit	96
Câu 103: Phần mềm Product Management - QDateTimeEdit	97

Câu 104: Phần mềm Quản lý bài hát - QTableWidgetItem - Design Time	99
Câu 105: Phần mềm Product Management – QTableWidgetItem runtime	100
Câu 106: Phần mềm Product Management – Pickle File	101
Câu 107: Phần mềm Student Management – Excel File	102
Câu 108: Phần mềm Music Management – XML File.....	103
Câu 109: Phần mềm Electronic Components Management – JSON File	104
Câu 110: Phần mềm Employee Management – CSV File.....	105
Câu 111: Chinook SQLite Database – QTableWidgetItem(**)	106
Câu 112: Phần mềm Product Management – SQLite Database(**)	108
MODULE 6	110
THƯ VIỆN TRONG PYTHON	110
Câu 113: Phân biệt Module, Package, Library	110
Câu 114: Trình bày cách cài đặt và sử dụng thư viện Numpy	111
Câu 115: Một số câu trắc nghiệm cơ bản liên quan tới thư viện Numpy	111
Câu 116: Các phép toán trên Mảng 1 chiều sử dụng Numpy	112
Câu 117: Thống kê cơ bản trên Mảng 1 chiều sử dụng Numpy - GUI	113
Câu 118: Trực quan hóa Mảng 1 chiều sử dụng Numpy - GUI	114
Câu 119: Tạo ma trận sử dụng Numpy - GUI	115
Câu 120: Các phép toán trên ma trận sử dụng Numpy.....	115
Câu 121: Các phép toán trên 2 ma trận sử dụng Numpy.....	117
Câu 122: Tính Gauss sử dụng Numpy.....	118
Câu 123: Trình bày cách cài đặt và sử dụng thư viện Pandas	120
Câu 124: Phân tích dữ liệu Employee - Pandas.....	120
Câu 125: Phân tích dữ liệu Sales - Pandas	121
Câu 126: Phân tích dữ liệu TCB - Pandas	121
Câu 127: Phân tích giá trị - Pandas.....	122
Câu 128: Xử lý dữ liệu rỗng - Pandas	123
Câu 129: Trình bày cách cài đặt và sử dụng thư viện Matplotlib & Seaborn	123
Câu 130: Trực quan hóa NetProfit với Matplotlib & Seaborn	124
Câu 131: Trực quan hóa PVD Assets với Matplotlib & Seaborn.....	125
Câu 132: Trực quan hóa cơ cấu doanh thu với Matplotlib & Seaborn vào PyQt6.....	126
Câu 133: Trực quan hóa mức lương với Matplotlib & Seaborn vào PyQt6.....	127
Câu 134: Re-practice Matplotlib & Seaborn vào PyQt6	128
Câu 135: Trình bày cách tích hợp Matplotlib & Seaborn vào PyQt6	128
Câu 136: Trình bày cách đóng gói và xuất bản phần mềm	129

MODULE 1

HÀM VÀ CÁC KỸ THUẬT XỬ LÝ HÀM

Nội dung kiến thức thực hành:

- + Khái niệm và ý nghĩa của hàm
- + Cấu trúc tổng quát của hàm.
- + Cách triệu gọi hàm
- + Cơ chế hoạt động của hàm
- + Phạm vi ảnh hưởng của biến trong hàm.
- + Xây dựng hàm với tham số mặc định.
- + Xây dựng hàm theo Lambda Expression.
- + Hàm trả về nhiều kết quả
- + Hàm đệ quy
- + Cách tạo hướng dẫn cho hàm.

Câu 1: Trình bày cơ chế hoạt động của hàm và cho ví dụ minh họa

Yêu cầu:

Sinh viên hãy trình bày cơ chế hoạt động của hàm và cho ví dụ minh họa về cơ chế hoạt động này.

Hướng dẫn:

Từ các kiến thức được giới thiệu trong môn học, Sinh viên cần tìm hiểu thêm trên mạng, các bài báo, diễn đàn để trả lời câu hỏi. Nên vẽ hình minh họa cơ chế hoạt động của hàm.

Câu 2: Phân biệt local variable và global variable

Yêu cầu:

Sinh viên hãy phân biệt local variable và global variable, cho ví dụ minh họa chi tiết.

Hướng dẫn:

Từ các kiến thức được giới thiệu trong môn học, Sinh viên cần tìm hiểu thêm trên mạng, các bài báo, diễn đàn để trả lời câu hỏi.

Câu 3: Cho biết kết quả xuất ra màn hình sau khi thực hiện mã lệnh

Yêu cầu:

Sinh viên hãy cho biết kết quả xuất ra màn hình sau khi thực hiện mã lệnh dưới đây, yêu cầu giải thích chi tiết vì sao ra kết quả này:

```
1 x=5
2 def fnX():
3     x=9
4     x+=2
5     print(f"x={x}")
6 fnX()
7 print(f"x={x}")
```

Hướng dẫn:

Từ các kiến thức được giới thiệu trong môn học, Sinh viên cần tìm hiểu thêm trên mạng, các bài báo, diễn đàn để trả lời câu hỏi.

Câu 4: Cho biết kết quả xuất ra màn hình sau khi thực hiện mã lệnh

Yêu cầu:

Sinh viên hãy cho biết quả xuất ra màn hình sau khi thực hiện mã lệnh dưới đây, yêu cầu giải thích chi tiết vì sao ra kết quả này:

```
1 x=5
2 def fnX():
3     global x
4     x=9
5     x+=2
6     print(f"x={x}")
7 fnX()
8 print(f"x={x}")
```

Hướng dẫn:

Từ các kiến thức được giới thiệu trong môn học, Sinh viên cần tìm hiểu thêm trên mạng, các bài báo, diễn đàn để trả lời câu hỏi.

Câu 5: Cho biết kết quả xuất ra màn hình sau khi thực hiện mã lệnh

Yêu cầu:

Sinh viên hãy cho biết quả xuất ra màn hình sau khi thực hiện mã lệnh dưới đây, yêu cầu giải thích chi tiết vì sao ra kết quả này:

```
1 def funX(m, n1=2, n2=4):
2     sum=n1+n2
3     for i in range(1, m):
4         sum=sum+i
5     return sum
6
7 s1 = funX(5)
8 print(f"s1 = {s1}")
9 s2 = funX(5,3)
10 print(f"s2 = {s2}")
11 s3 = funX(5,3,2)
12 print(f"s3 = {s3}")
```

Hướng dẫn:

Từ các kiến thức được giới thiệu trong môn học, Sinh viên cần tìm hiểu thêm trên mạng, các bài báo, diễn đàn để trả lời câu hỏi.

Câu 6: Trình bày cú pháp và cơ chế hoạt động của Lambda Expression

Yêu cầu:

Sinh viên hãy trình bày cú pháp và cơ chế hoạt động của Lambda Expression, cho một ví dụ minh họa.

Hướng dẫn:

Từ các kiến thức được giới thiệu trong môn học, Sinh viên cần tìm hiểu thêm trên mạng, các bài báo, diễn đàn để trả lời câu hỏi.

Câu 7: Cho biết kết quả xuất ra màn hình sau khi thực hiện mã lệnh

Yêu cầu:

Sinh viên hãy cho biết kết quả xuất ra màn hình sau khi thực hiện mã lệnh dưới đây, yêu cầu giải thích chi tiết vì sao ra kết quả này:

```
1 def handle(f,x,y):
2     if x % 2==0:
3         return f(x+2*y,x-y)
4     else:
5         return f(x-y,x+2*y)
6 def fn(x,y):
7     return x+y
8 z=handle(lambda x,y:fn(x,y),3,5)
9 print(f"z={z}")
10 t=handle(lambda x,y:fn(x,y),2,5)
11 print(f"t={t}")
12 k=handle(fn,6,8)
13 print(f"k={k}")
```

Hướng dẫn:

Từ các kiến thức được giới thiệu trong môn học, Sinh viên cần tìm hiểu thêm trên mạng, các bài báo, diễn đàn để trả lời câu hỏi.

Câu 8: Viết hàm tính điểm trung bình và xếp loại

Yêu cầu:

Viết hàm nhận vào 3 điểm: Toán, Lý và Hóa, thang điểm từ 0.0 tới 10.0. Hàm này trả về 2 giá trị:

- Điểm trung bình
- Xếp loại được chia theo: <5 (yếu), <7 (trung bình), <8 (khá), <9 (giỏi), còn lại là xuất sắc.

Sau đó viết các mã lệnh để cho phép người dùng nhập liên tục các điểm Toán, Lý, Hóa để kiểm thử các trường hợp khác nhau.

Hướng dẫn:

Từ các kiến thức được giới thiệu trong môn học, Sinh viên cần tìm hiểu thêm trên mạng, các bài báo, diễn đàn để trả lời câu hỏi.

Câu 9: Đệ quy là gì? Cho biết các điều kiện để lập trình đệ quy

Yêu cầu:

Đệ quy là gì? Cho biết các điều kiện để lập trình đệ quy.

Hướng dẫn:

Từ các kiến thức được giới thiệu trong môn học, Sinh viên cần tìm hiểu thêm trên mạng, các bài báo, diễn đàn để trả lời câu hỏi.

Câu 10: Trình bày và minh họa đệ quy tuyển tính

Yêu cầu:

Sinh viên trình bày và minh họa đệ quy tuyển tính bằng một ví dụ chi tiết.

Hướng dẫn:

Từ các kiến thức được giới thiệu trong môn học, Sinh viên cần tìm hiểu thêm trên mạng, các bài báo, diễn đàn để trả lời câu hỏi.

Câu 11: Trình bày và minh họa đệ quy đuôi

Yêu cầu:

Sinh viên trình bày và minh họa đệ quy đuôi bằng một ví dụ chi tiết.

Hướng dẫn:

Từ các kiến thức được giới thiệu trong môn học, Sinh viên cần tìm hiểu thêm trên mạng, các bài báo, diễn đàn để trả lời câu hỏi.

Câu 12: Trình bày và minh họa đệ quy nhị phân

Yêu cầu:

Sinh viên trình bày và minh họa đệ quy nhị phân bằng một ví dụ chi tiết.

Hướng dẫn:

Từ các kiến thức được giới thiệu trong môn học, Sinh viên cần tìm hiểu thêm trên mạng, các bài báo, diễn đàn để trả lời câu hỏi.

Câu 13: Trình bày và minh họa đệ quy đa tuyến

Yêu cầu:

Sinh viên trình bày và minh họa đệ quy đa tuyến bằng một ví dụ chi tiết.

Hướng dẫn:

Từ các kiến thức được giới thiệu trong môn học, Sinh viên cần tìm hiểu thêm trên mạng, các bài báo, diễn đàn để trả lời câu hỏi.

Câu 14: Trình bày và minh họa đệ quy lồng

Yêu cầu:

Sinh viên trình bày và minh họa đệ quy lồng bằng một ví dụ chi tiết.

Hướng dẫn:

Từ các kiến thức được giới thiệu trong môn học, Sinh viên cần tìm hiểu thêm trên mạng, các bài báo, diễn đàn để trả lời câu hỏi.

Câu 15: Trình bày và minh họa đệ quy tương hỗ

Yêu cầu:

Sinh viên trình bày và minh họa đệ quy tương hỗ bằng một ví dụ chi tiết.

Hướng dẫn:

Từ các kiến thức được giới thiệu trong môn học, Sinh viên cần tìm hiểu thêm trên mạng, các bài báo, diễn đàn để trả lời câu hỏi.

Câu 16: Viết hàm đệ quy chuyển cơ số 10 sang cơ số 2

Yêu cầu:

Từ bài học đệ quy tuyến tính, hãy cài đặt hàm chuyển cơ số 10 sang cơ số 2 và giải thích chi tiết cơ chế chạy đệ quy với các trường hợp:

(a) khi $n=14$

(c) khi $n=17$

Hướng dẫn:

Từ các kiến thức được giới thiệu trong môn học, Sinh viên cần tìm hiểu thêm trên mạng, các bài báo, diễn đàn để trả lời câu hỏi.

Câu 17: Viết hàm đệ quy chuyển cơ số 10 sang cơ số thập lục phân

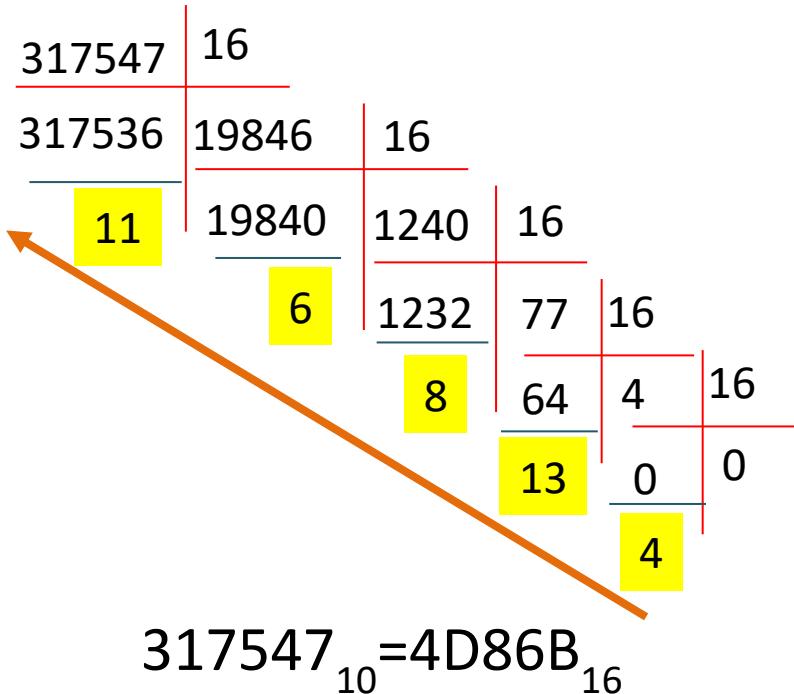
Yêu cầu:

Cho bảng chuyển đổi cơ số 10 sang cơ số thập lục phân:

Hệ cơ số 10	Hệ cơ số 16	Hệ cơ số 10	Hệ cơ số 16
0	0	8	8
1	1	9	9
2	2	10	A
3	3	11	B
4	4	12	C
5	5	13	D
6	6	14	E
7	7	15	F

Giả sử ta có số thập phân: 317547

Thì khi áp dụng toán học để chuyển qua cơ số 16 sẽ như sau:



Hãy viết hàm đệ quy chuyển cơ số 10 sang cơ số thập lục phân.

Hướng dẫn:

Từ các kiến thức được giới thiệu trong môn học, Sinh viên cần tìm hiểu thêm trên mạng, các bài báo, diễn đàn để trả lời câu hỏi.

Thử nghiệm với số 317547, và các số khác, cần giải thích chi tiết cơ chế chạy đệ quy của giải thuật này.

Câu 18: Viết hàm xử lý số Fibonacci (*)

Yêu cầu:

Dãy Fibonacci có dạng: $1 \rightarrow 1 \rightarrow 2 \rightarrow 3 \rightarrow 5 \rightarrow 8 \rightarrow 13 \rightarrow 21 \rightarrow 34 \rightarrow 55 \rightarrow 89 \dots$

Hãy thực hiện các yêu cầu sau:

- (a) Viết phương trình đệ quy cho dãy số Fibonacci ở trên theo dạng toán học
- (b) Viết hàm trả về số Fibonacci ở vị trí thứ x bất kỳ
- (c) Viết hàm trả về dãy số Fibonacci tính từ 1 cho tới x
- (d) Viết hàm khử đệ quy khi tính Fibonacci

Hướng dẫn:

Từ các kiến thức được giới thiệu trong môn học, Sinh viên cần tìm hiểu thêm trên mạng, các bài báo, diễn đàn để trả lời câu hỏi.

Câu (b) chỉ trả về 1 số Fibonacci duy nhất ở vị trí thứ x. Ví dụ như x=9 thì số Fibonacci là **34**.

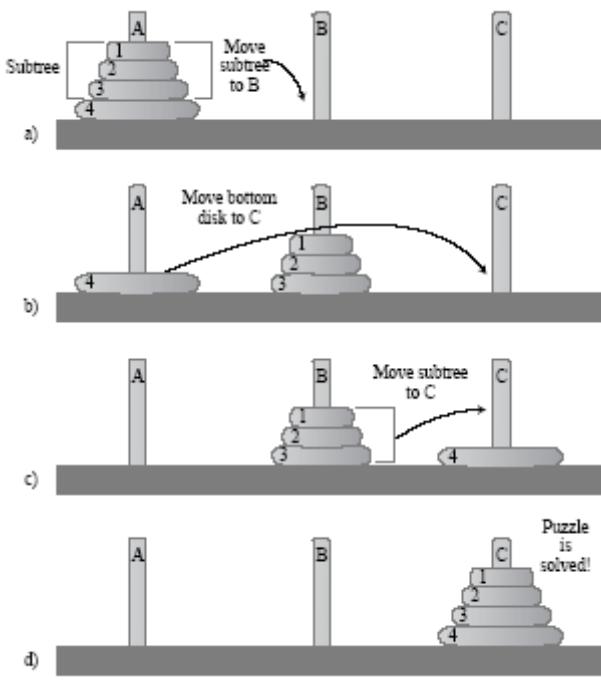
Câu (c) trả về danh sách số Fibonacci, ví dụ như x=9 thì kết quả là:
 $1 \rightarrow 1 \rightarrow 2 \rightarrow 3 \rightarrow 5 \rightarrow 8 \rightarrow 13 \rightarrow 21 \rightarrow 34$

Câu (d) Sinh viên dùng **vòng lặp** và các điều kiện **if elif** để khử đệ quy khi tính số Fibonacci.

Câu 19: Tháp Hà Nội (*)

Yêu cầu:

Hãy cài đặt giải thuật để quy định xử lý bài toán Tháp Hà Nội theo mô tả: Trò



chơi này gồm một bộ các đĩa kích thước khác nhau, có lỗ ở giữa, nằm xuyên trên ba cái cọc. Bài toán đó bắt đầu bằng cách sắp xếp các đĩa theo trật tự kích thước vào một cọc sao cho đĩa nhỏ nhất nằm trên cùng, tức là tạo ra một dạng hình nón. Yêu cầu của trò chơi là di chuyển toàn bộ số đĩa sang một cọc khác, tuân theo các quy tắc sau:

- ✓ Một lần Chỉ có 3 cột để di chuyển.
- ✓ Chỉ được di chuyển một đĩa trên cùng (không được di chuyển đĩa nằm giữa).
- ✓ Một đĩa chỉ có thể được đặt lên một đĩa lớn hơn (không nhất thiết hai đĩa này phải có kích thước liền kề, tức là đĩa nhỏ nhất có thể nằm trên đĩa lớn nhất).

Sau khi cài đặt hàm đệ quy tháp Hà Nội xong, thì Sinh viên cần phải giải thích chi tiết cơ chế đệ quy của việc thực hiện Tháp Hà Nội theo:

- Khi số đĩa = 2
- Khi số đĩa = 3
- Khi số đĩa = 4

Hướng dẫn:

Từ các kiến thức được giới thiệu trong môn học, Sinh viên cần tìm hiểu thêm trên mạng, các bài báo, diễn đàn để trả lời câu hỏi. Lưu ý phải giải thích cơ chế đệ quy bằng hình minh họa Stack giống như trong slide bài giảng. Sinh viên cần phải nắm rõ và hiểu sâu cơ chế chạy đệ quy.

Câu 20: Tạo hướng dẫn cho hàm

Yêu cầu:

Sinh viên hãy tạo hướng dẫn cho một số hàm trong bài học. Trình bày chi tiết cách tạo hướng dẫn cho hàm thông qua ví dụ cụ thể.

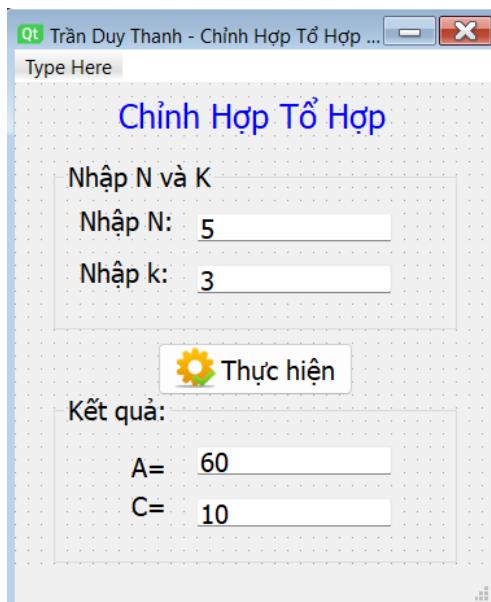
Hướng dẫn:

Từ các kiến thức được giới thiệu trong môn học, Sinh viên cần tìm hiểu thêm trên mạng, các bài báo, diễn đàn để trả lời câu hỏi.

Câu 21: Phần mềm tính chỉnh hợp - tổ hợp

Yêu cầu:

Sinh viên thiết kế giao diện phần mềm và viết các hàm để tính chỉnh hợp tổ hợp như dưới đây:



Người sử dụng nhập N và k, khi bấm “Thực hiện” chương trình sẽ tính chính hợp (A) và tổ hợp (C)

Hướng dẫn:

Từ các kiến thức được giới thiệu trong môn học, Sinh viên cần tìm hiểu thêm trên mạng, các bài báo, diễn đàn để trả lời câu hỏi. Xem lại kiến thức thiết kế giao diện cơ bản, và kỹ thuật xử lý sự kiện.

Câu 22: Viết hàm tính xác suất bóng đèn hư

Yêu cầu:

Bóng đèn được sản xuất ở công ty Thanh Thản được đóng gói theo hộp, mỗi hộp có **N** bóng đèn. Nhân viên chọn ngẫu nhiên **M** trong số **N** bóng đèn thuộc một hộp để kiểm tra. Giả sử hộp ấy có chứa **D** bóng đèn bị hư, tính xác suất để nhân viên đó lấy được 1 bóng đèn hư trong **M** bóng đèn được lấy.

-Sinh viên hãy viết hàm để hỗ trợ tính xác suất theo yêu cầu.

-Thiết kế giao diện tương tác người dùng phù hợp để kiểm thử kết quả.

Hướng dẫn:

Từ các kiến thức được giới thiệu trong môn học, Sinh viên cần tìm hiểu thêm trên mạng, các bài báo, diễn đàn để trả lời câu hỏi.

Và ta có thể tham khảo cách giải xác suất trên như sau:

Giả sử **N** = 12 bóng đèn

M=3 là số bóng đèn ngẫu nhiên mà nhân viên lấy trong 12 bóng đèn.

Giả sử bóng đèn hư **D**=5

Đặt **X** là số bóng đèn bị hư nhân viên đó lấy được trong 3 (**M**) bóng đèn. Khi đó, **X** tuân theo phân phối siêu bộ $H(N, D, M)$ tức là $H(12, 5, 3)$. Ta cần tìm $P(X=1)$ ta có:

$$P(X = 1) = \frac{C_D^1 C_{N-D}^{M-1}}{C_N^M} = \frac{C_5^1 C_7^2}{C_{12}^3} = 0.4773$$

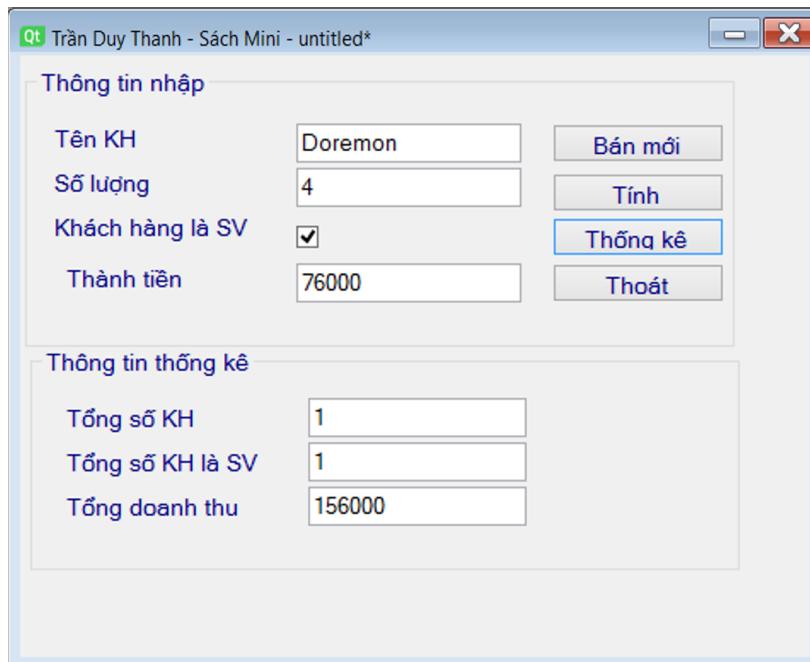
Dựa vào hướng dẫn ở trên, Sinh viên cần viết hàm để chương trình nhập vào các đối số tương ứng để xử lý theo công thức.

Khi thực nghiệm nên lấy ví dụ mẫu theo hướng dẫn này, sau đó bổ sung thêm các dữ liệu mẫu khác để thực nghiệm.

Câu 23: Phần mềm bán sách mini

Yêu cầu:

Sinh viên thiết kế phần mềm bán sách mini cho cửa hàng đồng giá Thanh Thản như dưới đây:



Khi chương trình vừa thực thi:

- +Tên Khách hàng không được phép nhập rỗng.
- +Số lượng sách là số nguyên dương
- +Mỗi quyển sách đều được bán với giá là 20000.
- Nút “Tính” tiên: Tính thành tiền và xuất kết quả lên Ô Thành tiền

Với Thành tiền = Số lượng * Đơn Giá

Lưu ý: Nếu Khách hàng là SV (Sinh viên) thì được giảm 5% thành tiền.

-Nút “Bán mới”:

+Xóa nội dung các control trong groupbox “Thông tin nhập”

+Đặt focus cho Textbox “Tên Khách Hàng”

-Nút “Thống Kê”: Tính và hiển thị kết quả trong groupbox “Thông tin Thống kê”: Cần phải thống kê rõ bao nhiêu khách hàng, bao nhiêu khách hàng là sinh viên, tổng giá trị

-Nút “Kết Thúc”: Thoát phần mềm

Hướng dẫn:

Từ các kiến thức được giới thiệu trong môn học, Sinh viên cần tìm hiểu thêm trên mạng, các bài báo, diễn đàn để trả lời câu hỏi.

Cần phải thực hiện các chức năng thông qua các hàm, tìm hiểu lại Widgets cơ bản để thiết kế giao diện, cũng như xử lý sự kiện người dùng.

Câu 24: Phần mềm Lucky Number (*)

Yêu cầu:

Thiết kế phần mềm Lucky Number có giao diện như dưới đây:



Khi chương trình vừa bắt đầu khởi động:

- 3 QLabel sẽ hiển thị 3 số 7.
- Máy tính có 100 đồng
- Người chơi có 100 đồng

Khi người chơi Click nút “quay số”:

Mỗi lần quay số, người chơi phải trả trước 30 đồng, tương ứng là máy sẽ được cộng thêm 30 đồng. Và nếu người chơi có số tiền ít hơn 30 đồng thì sẽ không được phép chơi tiếp.

Nếu mỗi lần quay được số 7, người chơi sẽ được thưởng như sau:

- Nếu ô đầu tiên là số 7 (random từ 0 đến 8), người chơi được thưởng 100 đồng + 50% số tiền của máy.
- Nếu ô số 2 là số 7 (random từ 0 đến 9), người chơi được thưởng 30 đồng + 50% tiền của máy.
- Nếu ô số 3 là số 7 (random từ 0 đến 10), người chơi được thưởng 10 đồng;
- Số tiền người chơi sẽ được cộng dồn nếu cả 3 ô cùng là số 7, tiền máy sẽ giảm đi phần trăm tương ứng

Nút “Game mới”

Cho phép 1 lược chơi mới hoàn toàn (set lại tiền người chơi và tiền máy là 100)

Nút “Kết thúc”

Gọi lệnh thoát phần mềm

Hướng dẫn:

Từ các kiến thức được giới thiệu trong môn học, Sinh viên cần tìm hiểu thêm trên mạng, các bài báo, diễn đàn để trả lời câu hỏi. Cần biết kết hợp sử dụng số ngẫu nhiên và cấu trúc điều kiện với giao diện.

MODULE 2

CẤU TRÚC DỮ LIỆU TRONG PYTHON

Nội dung kiến thức thực hành:

- Thực hành với dữ liệu Chuỗi
 - + Cấu trúc chuỗi
 - + Và các hàm xử lý chuỗi
- Kiểu dữ liệu Mảng và các thao tác trên mảng
- Kiểu dữ liệu Tập Hợp:
 - + List và các thao tác trên List
 - + Tuple và các thao tác trên Tuple
 - + Set và các thao tác trên Set
 - + Dictionary và các thao tác trên Dictionary

Câu 25: Viết chương trình kiểm tra chuỗi đối xứng

Yêu cầu:

Sinh viên sử dụng vòng lặp while vô tận, cho phép Nhập vào một Chuỗi → Xuất Chuỗi này có phải đối xứng hay không? Hỏi người sử dụng có tiếp tục phần mềm. Nếu tiếp tục thì nhập Chuỗi mới, còn không thì thoát và thông báo cảm ơn.

Sinh viên cần nâng cấp bài này bằng cách thiết kế giao diện tương tác PyQt6 hoặc Tkinter.

Hướng dẫn:

Từ các kiến thức được giới thiệu trong môn học, Sinh viên cần tìm hiểu thêm trên mạng, các bài báo, diễn đàn để trả lời câu hỏi. Ngoài ra sinh viên có thể tham khảo mã lệnh mẫu dưới đây:

```
1 def CheckDoiXung(s):
2     flag=True
3     for i in range(len(s)):
4         if s[i]!=s[len(s)-i-1]:
5             flag=False
6             break
7     return flag
8 def main():
9     print("Nhập 1 chuỗi:")
10    s=input()
11    if(CheckDoiXung(s)):
12        print("Chuỗi bạn nhập đối xứng")
13    else:
14        print("Chuỗi bạn nhập không đối xứng")
15    while True:
16        main()
17        print("Tiếp không Thím?(c/k):",end=' ')
18        s=input()
19        if s=="k":
20            break
21    print("CÁM ƠN THÍM")
```

Câu 26: Các hàm xử lý chuỗi cơ bản

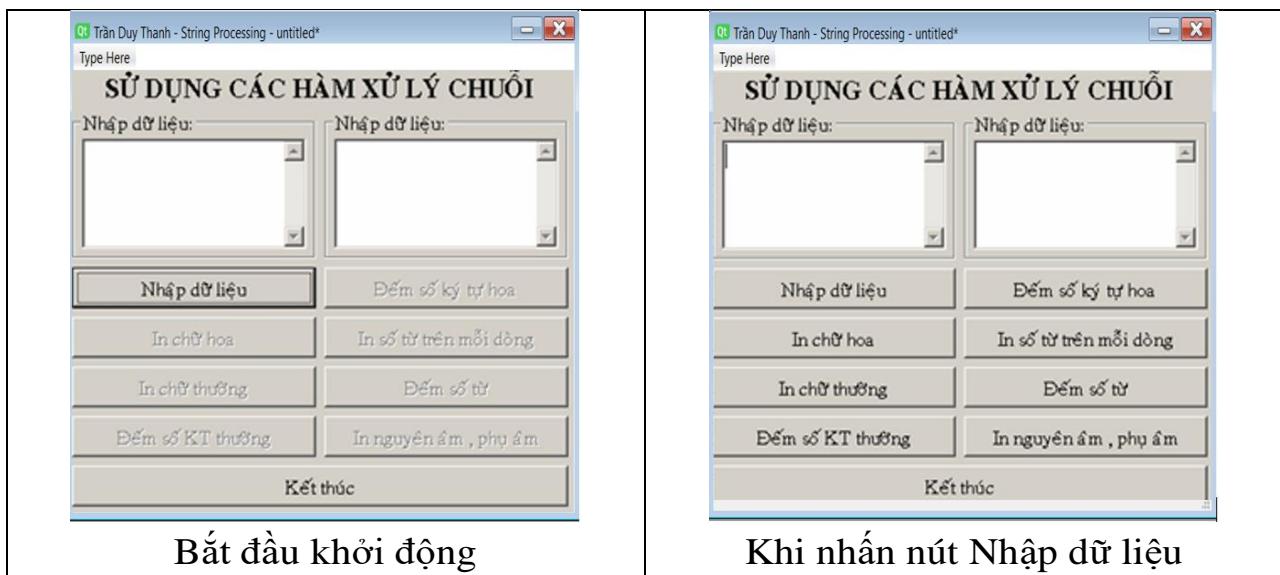
Yêu cầu:

Viết chương trình cho phép nhập vào 1 chuỗi. Yêu cầu xuất ra:

- Bao nhiêu chữ IN HOA
- Bao nhiêu chữ in thường
- Bao nhiêu chữ là chữ số
- Bao nhiêu chữ là ký tự đặc biệt
- Bao nhiêu chữ là khoảng trắng
- Bao nhiêu chữ là Nguyên Âm

- Bao nhiêu chữ là Phụ âm

Cần thiết kế giao diện tương tự như dưới đây để thực nghiệp kết quả:



Ô nhập liệu bên trái là ô để người dùng nhập một chuỗi bất kỳ, ô dữ liệu bên phải là ô hiển thị kết quả.

Hướng dẫn:

Từ các kiến thức được giới thiệu trong môn học, Sinh viên cần tìm hiểu thêm trên mạng, các bài báo, diễn đàn để trả lời câu hỏi.

Câu 27: Viết chương trình tối ưu chuỗi danh từ

Yêu cầu:

Một chuỗi Danh Từ được gọi là tối ưu khi: Không chứa các khoảng trắng dư thừa, các từ cách nhau bởi một khoảng trắng, ký tự đầu tiên của các từ viết Hoa.

- Ví dụ đầu vào ta có 1 chuỗi “ trẦN THanh THẦN ” thì sau khi thực hiện chương trình sẽ ra chuỗi kết quả tối ưu “Trần Thanh Thần”.

- Sinh viên **cần thiết kế giao diện** để người dùng tương tác, cung cấp chức năng tối ưu chuỗi danh từ trên giao diện để kiểm thử.

Hướng dẫn:

Có rất nhiều phương pháp để làm bài này, Sinh viên có thể áp dụng phối hợp các hàm như `join()`, `split()` và `title()` để thực hiện bài này.

Dưới đây là mã lệnh minh họa chức năng tối ưu chuỗi danh từ:

```
1 def ToiUuChuoi(s):
2     return " ".join(s.split()).title()
3
4 s="        trẦN      THanh       THẦN      "
5 print(s,"=>",len(s))
6 s=ToiUuChuoi(s)
7 print(s,"=>",len(s))
```

Kết quả:

```
Run App ×
C:\Python\Python312\python.exe E:\Elearning\ProgrammingEx\Ex26\App.py
        trẦN      THanh       THẦN      => 40
Trần Thanh Thần => 15
```

Câu 28: Thống kê số trong chuỗi

Yêu cầu:

Cho 1 Chuỗi có định dạng như sau “5;7;8;-2;8;11;13;9;10” (có thể nhập bất kỳ từ bàn phím). Sinh viên hãy viết các mã lệnh để thực hiện:

- Xuất các chữ số trên các dòng riêng biệt
- Thống kê có bao nhiêu chữ số chẵn
- Thống kê có bao nhiêu số âm
- Thống kê có bao nhiêu chữ số nguyên tố
- Tính giá trị trung bình

Hướng dẫn:

Tách chuỗi theo ký tự “;” rồi ép kiểu từng phần tử về kiểu số để kiểm tra.
Hoặc Sinh viên có thể dùng giải pháp tùy ý.

Câu 29: Trích lọc số âm trong chuỗi

Yêu cầu:

Viết một hàm đặt tên là **NegativeNumberInStrings(s)**. Hàm này có đối số truyền vào là một chuỗi bất kỳ, Hãy viết lệnh để xuất ra các số nguyên âm trong chuỗi.

Ví dụ: Nếu nhập vào chuỗi “abc-**5xyz-12k9l--p**” thì hàm phải xuất ra được 2 số nguyên âm đó là -5 và -12

Hướng dẫn:

Từ các kiến thức được giới thiệu trong môn học, Sinh viên cần tìm hiểu thêm trên mạng, các bài báo, diễn đàn để trả lời câu hỏi.

Sinh viên có thể cung cấp giao diện tương tác người dùng để kiểm thử các dữ liệu đầu vào khác nhau.

Câu 30: Tách tên bài hát và loại tập tin

Yêu cầu:

Cho một chuỗi là đường dẫn của 1 file nhạc, ví dụ: d:\music\muabui.mp3

Hãy viết 2 hàm để:

- Lấy ra muabui.mp3
- Lấy ra muabui

Lưu ý đường dẫn bài hát là bất kỳ. Nên khi truyền vào bài hát nào thì lấy chính xác theo bài hát đó.

Hướng dẫn:

Từ các kiến thức được giới thiệu trong môn học, Sinh viên cần tìm hiểu thêm trên mạng, các bài báo, diễn đàn để trả lời câu hỏi.

Khi ý các dấu ký tự đặc biệt \ hay \\ hay /

Câu 31: Tách lấy tên các thành phần trong URL

Yêu cầu:

Cho biết cấu trúc của một URL như sau:



Sinh viên hãy viết hàm nhận vào 1 chuỗi là URL bất kỳ có cấu trúc như trên, kết quả sau khi thực hiện của chương trình là trả về các thành phần của URL này.

Hướng dẫn:

Từ các kiến thức được giới thiệu trong môn học, Sinh viên cần tìm hiểu thêm trên mạng, các bài báo, diễn đàn để trả lời câu hỏi.

Câu 32: Mô tả mảng và các thao tác trên mảng

Yêu cầu:

Sinh viên mô tả mảng và các thao tác trên mảng. Thông thường với cấu trúc dữ liệu mảng khi ta sử dụng thì thường nó bao gồm các chức năng chính: Khởi tạo, lưu trữ, hiển thị dữ liệu, sửa dữ liệu, xóa dữ liệu, tìm kiếm và sắp xếp.

Hướng dẫn:

Từ các kiến thức được giới thiệu trong môn học, Sinh viên cần tìm hiểu thêm trên mạng, các bài báo, diễn đàn để trả lời câu hỏi.

Câu 33: Xây dựng phần mềm tương tác mảng

Yêu cầu:

Thiết kế phần mềm tương tác Mảng bằng giao diện người dùng PyQt6 – Qt Designer như dưới đây:



- “Xuất mảng ngẫu nhiên”: Khi người sử dụng nhấn nút này thì mảng 10 phần tử gồm các giá trị ngẫu nhiên sẽ được tạo ra và hiển thị vào ô Mảng Gốc.
- Tất cả các chức năng còn lại tương ứng với các nhãn được viết trong QPushButton, người sử nhấn button nào thì chương trình sẽ thực thi các mã lệnh và hiển thị kết quả này vào ô Kết quả.

Hướng dẫn:

Từ các kiến thức được giới thiệu trong môn học, Sinh viên cần tìm hiểu thêm trên mạng, các bài báo, diễn đàn để trả lời câu hỏi.

Câu 34: Các thao tác trên mảng

Yêu cầu:

Viết chương trình nhập vào một mảng số tự nhiên có n phần tử. Hãy xuất ra màn hình:

- Dòng 1 : gồm các số lẻ, tổng cộng có bao nhiêu số lẻ.
- Dòng 2 : gồm các số chẵn, tổng cộng có bao nhiêu số chẵn.
- Dòng 3 : gồm các số nguyên tố.
- Dòng 4 : gồm các số không phải là số nguyên tố.
- Dòng 5 : mảng sau khi xóa các số chẵn

Hướng dẫn:

Từ các kiến thức được giới thiệu trong môn học, Sinh viên cần tìm hiểu thêm trên mạng, các bài báo, diễn đàn để trả lời câu hỏi.

Câu 35: Mô tả và phân biệt cấu trúc dữ liệu tập hợp

Yêu cầu:

Sinh viên hãy mô tả các cấu trúc dữ liệu tập hợp trong Python:

- List
- Tuple
- Set
- Dictionary

Sau đó hay so sánh ưu điểm và nhược điểm đối với các loại cấu trúc này.

Hướng dẫn:

Từ các kiến thức được giới thiệu trong môn học, Sinh viên cần tìm hiểu thêm trên mạng, các bài báo, diễn đàn để trả lời câu hỏi.

Câu 36: Xác định kết quả khi thực thi List

Yêu cầu:

Cho một list có dữ liệu như dưới đây

lst = [3, 0, 1, 5, 2]

x = 2

Hãy cho biết kết quả sau khi thực hiện các lệnh dưới đây:

- (a) lst[0]?
- (b) lst[3]?
- (c) lst[x]?
- (d) lst[-x]?
- (e) lst[x + 1]?
- (f) lst[x] + 1?
- (g) lst[lst[x]]?
- (h) lst[lst[lst[x]]]?

Hướng dẫn:

Từ các kiến thức được giới thiệu trong môn học, Sinh viên cần tìm hiểu thêm trên mạng, các bài báo, diễn đàn để trả lời câu hỏi.

Câu 37: Xác định kết quả khi thực thi List

Yêu cầu:

Cho một list có dữ liệu như dưới đây

lst = [20, 1, -34, 40, -8, 60, 1, 3]

Hãy cho biết kết quả sau khi thực hiện các lệnh dưới đây:

- (a) lst
- (b) lst[0:3]

- (c) lst[4:8]
- (d) lst[4:33]
- (e) lst[-5:-3]
- (f) lst[-22:3]
- (g) lst[4:]
- (h) lst[:]
- (i) lst[:4]
- (j) lst[1:5]
- (k) -34 in lst
- (l) -34 not in lst
- (m) len(lst)

Hướng dẫn:

Từ các kiến thức được giới thiệu trong môn học, Sinh viên cần tìm hiểu thêm trên mạng, các bài báo, diễn đàn để trả lời câu hỏi.

Câu 38: Các thao tác trên List

Yêu cầu:

Viết chương trình cho phép:

- Khởi tạo list với ngẫu nhiên 10 phần tử có giá trị từ -100 tới 100
- Thêm phần tử vào list
- Nhập k, kiểm tra k xuất hiện bao nhiêu lần trong list
- Tính tổng các số hoàn thiện trong list
- Sắp xếp: Tăng, Giảm
- Xóa list: Xóa một phần tử, Xóa các số âm, Xóa toàn bộ list

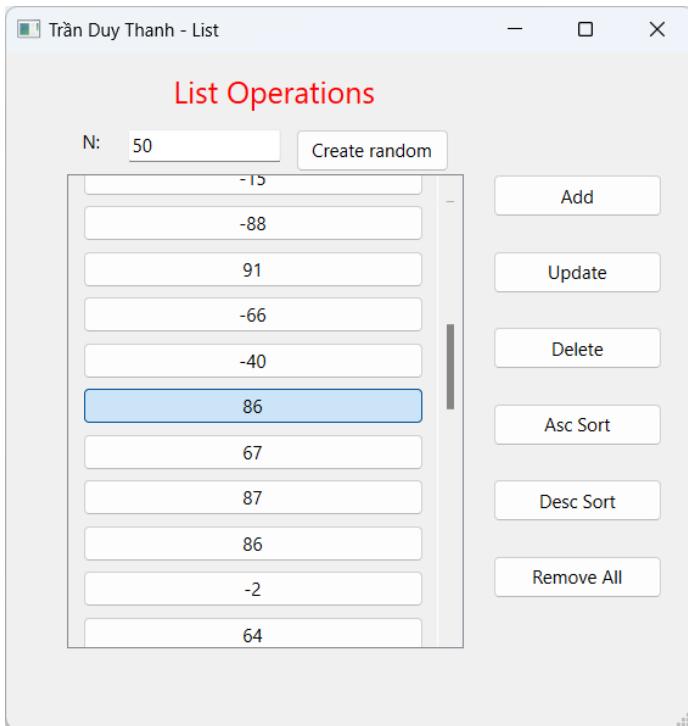
Hướng dẫn:

Từ các kiến thức được giới thiệu trong môn học, Sinh viên cần tìm hiểu thêm trên mạng, các bài báo, diễn đàn để trả lời câu hỏi.

Câu 39: Phần mềm thao tác trên List

Yêu cầu:

Thiết kế phần mềm với giao diện PyQt6 – Qt Design như dưới đây:



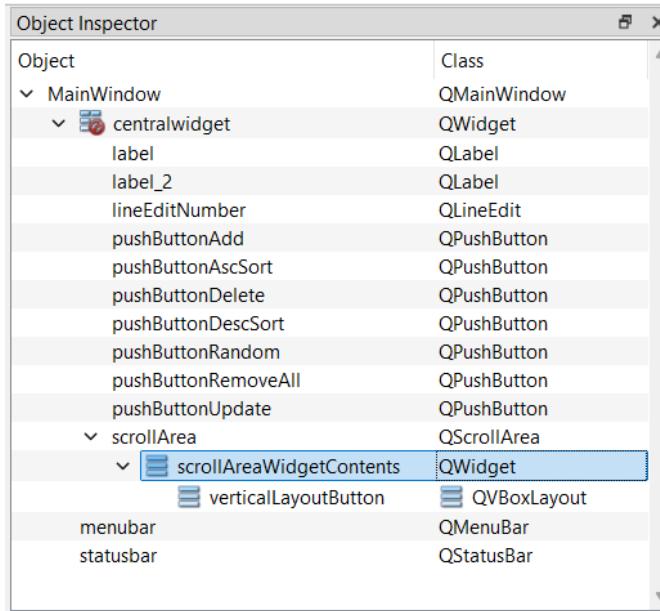
Yêu cầu chức năng như sau:

- “Create random”: Chương trình sẽ tạo N đối tượng QPushButton với các nhãn là các số ngẫu nhiên từ -100 tới 100 lên giao diện.
- “Add”: Khi chọn Button này thì chương trình sẽ chèn thêm 1 QPushButton ở cuối giao danh sách Button với nhãn ngẫu nhiên [0..10]
- “Update”: Khi nhấn vào Button này thì QPushButton nào đang chọn trên giao diện sẽ đổi nhãn thành giá trị nhãn//10. Ví dụ số 86 đổi thành 86//10=8
- “Delete”: QPushButton là số âm sẽ bị xóa khỏi danh sách
- “Desc Sort”: Sắp xếp danh sách QPushButton tăng dần theo nhãn
- “Asc Sort”: Sắp xếp danh sách QPushButton giảm dần theo nhãn
- “Remove all”: Xóa toàn bộ QPushButton ra khỏi giao diện

Hướng dẫn:

Từ các kiến thức được giới thiệu trong môn học, Sinh viên cần tìm hiểu thêm trên mạng, các bài báo, diễn đàn để trả lời câu hỏi.

Gói ý cấu trúc giao diện mẫu của bài tập:



Dưới đây là coding mẫu tạo QPushButton ngẫu nhiên khi nhấn vào nút “Create Random”:

```
1 def clearLayout(self,layout):
2     if layout is not None:
3         while layout.count():
4             item = layout.takeAt(0)
5             widget = item.widget()
6             if widget is not None:
7                 widget.deleteLater()
8             else:
9                 self.clearLayout(item.layout())
10    def createRandom(self):
11        self.clearLayout(self.verticalLayoutButton)
12        n=int(self.lineEditNumber.text())
13        for i in range(n):
14            x=random.randint(-100,100)
15            print(x)
16            title=f"{x}"
17            btn=QPushButton(text=title)
18            self.verticalLayoutButton.addWidget(btn)
```

Các câu khác Sinh viên tự suy luận, các câu sắp xếp thì clear giao diện rồi nạp lại.

Câu 40: Xử lý List đa chiều

Yêu cầu:

Viết chương trình cho phép xử lý List đa chiều trong Python:

- Khởi tạo và nhập vào ma trận MxN phần tử ngẫu nhiên
- Xuất dòng bất kỳ nhập từ bàn phím
- Xuất cột bất kỳ từ bàn phím
- Xuất số MAX trong ma trận
- Xuất số MIN trong ma trận
- Sắp xếp Ma trận theo dòng dạng zic zắc: Dòng chẵn sắp xếp tăng dần, dòng lẻ sắp xếp giảm dần

Hướng dẫn:

Từ các kiến thức được giới thiệu trong môn học, Sinh viên cần tìm hiểu thêm trên mạng, các bài báo, diễn đàn để trả lời câu hỏi.

Câu 41: Viết hàm thống kê sử dụng Tuple

Yêu cầu:

Hãy viết một hàm thống kê, hàm này nhận vào một danh sách các con số và sử dụng Tuple để trả về 3 kết quả: số lượng danh sách, tổng các phần tử trong danh sách, giá trị trung bình, số lớn nhất, số nhỏ nhất.

Viết các mã lệnh để kiểm thử hàm thống kê ở trên.

Hướng dẫn:

Từ các kiến thức được giới thiệu trong môn học, Sinh viên cần tìm hiểu thêm trên mạng, các bài báo, diễn đàn để trả lời câu hỏi. Coding minh họa:

```
1 | def calculate_statistics(*numbers):  
2 |     count = len(numbers)  
3 |     total = sum(numbers)
```

```
4     mean = total / count
5     maxValue=max(numbers)
6     minValue=min(numbers)
7     return count, total, mean,maxValue,minValue
8
9 stats = calculate_statistics(10, 20, 30, 40, 50)
10 print(stats)
11 (count,total,mean,maxValue,minValue)=stats
12 print(f"count={count}")
13 print(f"total={count}")
14 print(f"average={mean}")
15 print(f"maxValue={maxValue}")
16 print(f"minValue={minValue}")
```

Kết quả:

```
Run Create new Qt Designer App (2)
C:\Python\Python312\python.exe
(5, 150, 30.0, 50, 10)
count=5
total=5
average=30.0
maxValue=50
minValue=10
```

Câu 42: Tìm các phần tử duy nhất trong một list có chứa các Tuple

Yêu cầu:

Cho một list **data** chứa các tuple, bạn cần viết một chương trình để trả về một danh sách chứa các phần tử duy nhất từ các tuple trong data. Ví dụ:

```
data = [(1, 2), (2, 3), (3, 4), (1, 2)]
```

Kết quả mong đợi khi chạy chương trình: [1, 2, 3, 4]

Hướng dẫn:

- Sử dụng một **set** để lưu trữ các phần tử duy nhất.
- Duyệt qua từng tuple trong list **data**, sau đó duyệt qua từng phần tử trong tuple để thêm vào set.

Câu 43: Sắp xếp Sinh viên theo nhiều tiêu chí

Yêu cầu:

Cho một danh sách **students** chứa các tuple, mỗi tuple chứa thông tin về một sinh viên (tên, điểm số, tuổi). Sinh viên cần sắp xếp danh sách này theo điểm số giảm dần. Nếu có các sinh viên có cùng điểm số, hãy sắp xếp theo tuổi tăng dần.

Ví dụ:

```
students = [('Phúc', 85, 20), ('Phan', 75, 22), ('Thanh', 85, 21), ('Thản', 80, 21)]
```

Kết quả mong đợi:

```
[('Phúc', 85, 20), ('Thanh', 85, 21), ('Thản', 80, 21), ('Phan', 75, 22)]
```

Hướng dẫn:

-Sử dụng hàm sorted() và truyền vào một lambda function để chỉ định tiêu chí sắp xếp.

-Lambda function này có thể trả về một tuple (điểm số, tuổi) để sắp xếp danh sách theo hai tiêu chí.

-Hoặc thích dùng cách gì cũng được, miễn sao sắp xếp đúng yêu cầu

Câu 44: Tính tổng các phần tử nhiều tiêu chí

Yêu cầu:

Cho một danh sách **numbers_list** chứa các tuple, mỗi tuple chứa các số nguyên. Sinh viên hãy tính tổng của các phần tử có giá trị chẵn nằm ở vị trí lẻ trong mỗi tuple và trả về tổng này. Ví dụ:

```
numbers_list = [(1, 2, 3, 4), (5, 13, 7, 8), (9, 10, 11, 12)]
```

Kết quả mong đợi là 36 [tổng của 6 (vị trí 1,3), 8 (vị trí 3), và 22 (vị trí 1,3)]

Lưu ý Sinh viên cần viết hàm để xử lý, và các mã lệnh để kiểm thử hàm này.

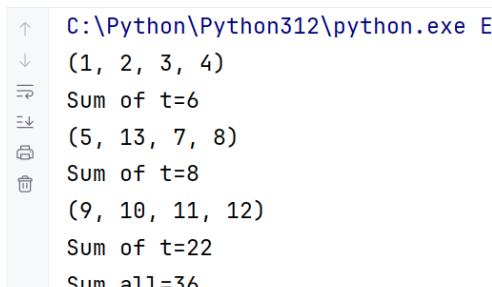
Hướng dẫn:

- Sử dụng một vòng lặp để duyệt qua từng tuple trong danh sách numbers_list.
- Sử dụng index để truy cập các phần tử ở vị trí lẻ (index \% 2 == 1) và vị trí này phải có giá trị chẵn
- Tính tổng các phần tử này và lưu kết quả vào một biến, sau đó trả về kết quả cuối cùng.

Coding gợi ý:

```
1 numbers_list = [(1, 2, 3, 4), (5, 13, 7, 8), (9, 10, 11, 12)]
2 sumAll=0
3 for t in numbers_list:
4     print(t)
5     sumt=0
6     for v in t:
7         index=t.index(v)
8         if index %2==1 and v %2==0:
9             sumt+=v
10    print(f"Sum of t={sumt}")
11    sumAll+=sumt
12 print(f"Sum all={sumAll}")
```

Kết quả minh họa:



```
C:\Python\Python312\python.exe E
(1, 2, 3, 4)
Sum of t=6
(5, 13, 7, 8)
Sum of t=8
(9, 10, 11, 12)
Sum of t=22
Sum all=36
```

Câu 45: Phần mềm Quản lý sách (*)

Yêu cầu:

Kết hợp List và Tuple để quản lý thông tin sách: Mỗi cuốn sách gồm có các thông tin: ISBN, Title, Authors, PublishYear, Publisher

Sinh viên hãy sử dụng cấu trúc dữ liệu phù hợp để cung cấp các tính năng:

-Thêm sách

-Chỉnh sửa thông tin sách

-Xóa sách theo ISBN

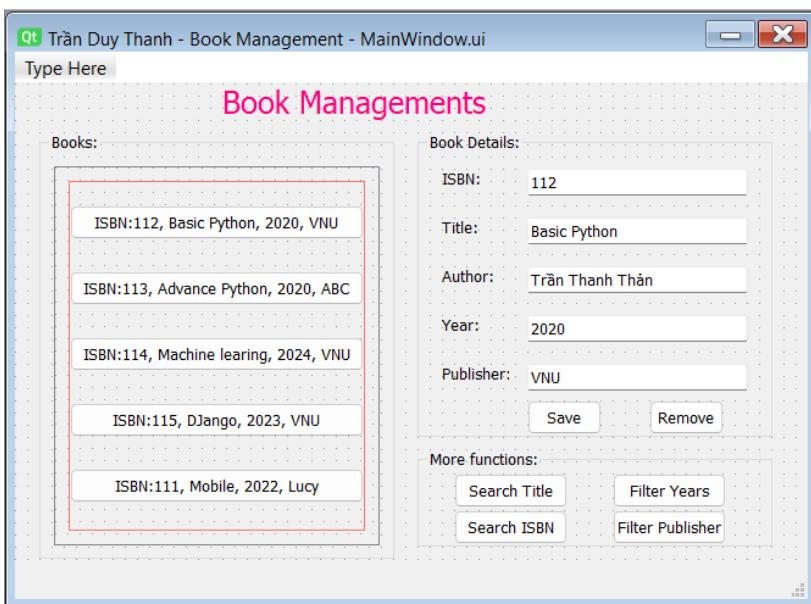
-Duyệt danh sách các cuốn sách

-Tìm sách theo Title

-Lọc sách theo năm xuất bản (chọn thời gian X và thời gian Y để lọc)

-Lọc sách theo nhà xuất bản

-Nâng cao (*): Sau khi thực nghiệm các chức năng trong console, thì Sinh viên cần thiết kế giao diện PyQt6 – QT Design phù hợp để người dùng có thể sử dụng phần mềm quản lý sách:



+ Danh sách các Book được hiển thị trong mục Books, mỗi book đại diện là QPushButton. Khi người dùng chọn Sách trong Books bên màn hình bên trái thì thông tin chi tiết sẽ hiển thị trong Book Details

+ Button “Save”: Vừa lưu mới vừa cập nhật, nếu ISBN tồn tại thì cập nhật.

+ Button “Remove”: Xóa book đang chọn

+ Các chức năng mở rộng như: Search Title để tìm sách theo tiêu đề sách, Filter Years lọc các cuốn sách theo năm, Search ISBN để tìm sách theo mã số ISBN, và Filter Publisher để lọc sách theo nhà xuất bản. Sinh viên để xuất giao diện con phù hợp để thực hiện các chức năng này.

Hướng dẫn:

Từ các kiến thức được giới thiệu trong môn học, Sinh viên cần tìm hiểu thêm trên mạng, các bài báo, diễn đàn để trả lời câu hỏi.

Câu 46: Tìm tập hợp con lớn nhất không có phần tử chung với tập hợp ban đầu**Yêu cầu:**

Cho một tập hợp **main_set** và một danh sách các tập hợp con **subsets**. Sinh viên cần viết một Hàm có 2 đối số để tìm tập hợp con lớn nhất từ **subsets** mà không có bất kỳ phần tử nào chung với **main_set**. Ví dụ:

main_set = {1, 2, 3, 4}

subsets = [{2, 3}, {1, 2}, {5, 6}, {3, 4}]

Thì sau khi thực hiện thì Kết quả mong đợi là {5, 6} vì {5, 6} là tập hợp con lớn nhất không có phần tử nào chung với {1, 2, 3, 4}.

Hướng dẫn:

- Sử dụng cấu trúc **set** để tìm tập hợp con lớn nhất không có phần tử chung với **main_set**.
- Duyệt qua từng tập hợp con trong **subsets**, kiểm tra xem có phần tử nào chung với **main_set** hay không.
- Lưu tập hợp con đó nếu không có phần tử chung và nếu lớn hơn các tập hợp khác đã kiểm tra.

Câu 47: Kiểm tra tính đối xứng của một tập hợp các tuple**Yêu cầu:**

Cho một tập hợp **set_of_tuples**, Sinh viên viết hàm kiểm tra xem tập hợp này có chứa các tuple đối xứng hay không. Một tuple được coi là đối xứng nếu có một tuple khác trong tập hợp mà khi đảo ngược thứ tự các phần tử của nó, ta thu được tuple ban đầu. Ví dụ:

set_of_tuples = {(1, 2), (2, 1), (3, 4), (4, 3), (5, 6)}

Thì sau khi thực hiện thì Kết quả mong đợi là True vì (1, 2) và (2, 1), cũng như (3, 4) và (4, 3) là các cặp tuple đối xứng.

Hướng dẫn:

- Sử dụng một set để lưu trữ các tuple đã kiểm tra.
- Duyệt qua từng tuple trong set_of_tuples, kiểm tra xem tuple đảo ngược của nó đã có trong set hay chưa.
- Nếu đã có, tuple đó là đôi xứng; nếu chưa, thêm tuple ban đầu và tuple đảo ngược vào set và tiếp tục duyệt.

Câu 48: Tìm tất cả các phần tử là hợp của các tập hợp con trong một danh sách các set

Yêu cầu:

Cho một danh sách sets_list chứa các set, Sinh viên cần viết hàm tìm tất cả các phần tử mà là phần tử của ít nhất một set trong danh sách này. Ví dụ:

```
sets_list = [{1, 2, 3}, {2, 3, 4}, {3, 4, 5}]
```

Sau khi thực hiện thì Kết quả mong đợi là {1, 2, 3, 4, 5} vì các phần tử này là hợp của các tập hợp con trong ‘sets_list’.

Hướng dẫn:

- Sử dụng phép toán set để lưu trữ tất cả các phần tử từ các set trong sets_list.
- Sử dụng phép toán union() hoặc update() để hợp nhất các set trong sets_list thành một set duy nhất.
- Kết quả là tất cả các phần tử của set này.

Câu 49: Tạo dictionary từ danh sách các tuple và tính tổng các giá trị trùng lặp

Yêu cầu:

Cho một danh sách data chứa các tuple, mỗi tuple có hai phần tử: một key và một giá trị số nguyên. Sinh viên hãy viết một hàm để tạo một dictionary từ các tuple này và tính tổng các giá trị nếu các key trùng nhau. Ví dụ:

data = [('a', 10), ('b', 20), ('a', 30), ('c', 40)]

Thì sau khi thực thi Kết quả mong đợi là {'a': 40, 'b': 20, 'c': 40} vì 'a' xuất hiện hai lần với tổng giá trị là 40.

Hướng dẫn:

- Sử dụng một dictionary để lưu trữ các key và giá trị tương ứng.
- Duyệt qua từng tuple trong danh sách data, kiểm tra xem key đã có trong dictionary chưa.
- Nếu đã có, cộng giá trị mới vào giá trị hiện tại của key đó; nếu chưa, thêm key vào dictionary với giá trị là giá trị của tuple đó.

Câu 50: Sắp xếp dictionary theo giá trị và giữ nguyên key nếu giá trị trùng nhau

Yêu cầu:

Cho một dictionary scores lưu trữ điểm số của các sinh viên (key là tên sinh viên và value là điểm số). Sinh viên hãy viết hàm sắp xếp dictionary này theo giá trị điểm số giảm dần. Nếu có sinh viên có cùng điểm số, giữ nguyên thứ tự của chúng. Ví dụ:

scores = {'Phan': 85, 'Phúc': 75, 'Thanh': 85, 'Thản': 80}

Sau khi thực thi chương trình thì Kết quả mong đợi là {'Phan': 85, 'Thanh': 85, 'Thản': 80, 'Phúc': 75} với 'Phan' và 'Thanh' có cùng điểm số.

Hướng dẫn:

- Sử dụng hàm sorted() và truyền vào một lambda function để sắp xếp dictionary theo giá trị.
- Lambda function này có thể trả về một tuple (value, key) để sắp xếp dictionary theo giá trị và giữ nguyên thứ tự nếu các giá trị bằng nhau.

Câu 51: Tạo dictionary chứa thông tin thống kê từ một danh sách các dictionary

Yêu cầu:

Cho một danh sách **students_data** chứa các dictionary, mỗi dictionary chứa thông tin về một sinh viên (tên, tuổi, điểm số). Sinh viên hãy viết một chương trình để tạo một dictionary ‘statistics’ chứa thông tin thống kê về điểm số trung bình, tuổi trung bình của các sinh viên. Ví dụ:

```
students_data = [  
    {'name': 'Phúc', 'age': 20, 'score': 85},  
    {'name': 'Thanh', 'age': 22, 'score': 75},  
    {'name': 'Thản', 'age': 21, 'score': 90}  
]
```

Thì sau khi thực hiện Kết quả mong đợi là {‘average_score’: 83.3, ‘average_age’: 21} với điểm số trung bình là 83.3 và tuổi trung bình là 21.

Hướng dẫn:

- Khởi tạo một dictionary ‘statistics’ có các key là ‘average_score’ và ‘average_age’.
- Duyệt qua từng dictionary trong students_data, tính tổng điểm số và tổng tuổi.
- Sau đó, tính điểm số trung bình và tuổi trung bình và lưu vào dictionary statistics.

MODULE 3

LẬP TRÌNH HƯỚNG ĐỐI TƯỢNG

Nội dung kiến thức thực hành:

- Các khái niệm về Lập trình hướng đối tượng
- Các quy tắc khai báo và sử dụng lớp-đối tượng
- Các loại phương thức
- Tham chiếu self
- Overloading Method và Overriding Method
- Alias và cơ chế gom rác tự động
- Cách khai báo phương thức tĩnh và ý nghĩa sử dụng
- Kỹ thuật Ké thừa và cách sử dụng
- Thực hành một số kiểu class đặc biệt: Kiểu Enums, Singleton class, Wrapper class, Inner class

Câu 52: Trình bày Khái niệm về lập trình hướng đối tượng

Yêu cầu:

Trình bày khái niệm về lập trình hướng đối tượng. Trong đó hãy mô tả ý nghĩa khái quát về: Object, Class, Inheritance, Polymorphism, Abstraction và Encapsulation.

Hướng dẫn:

Từ các kiến thức được giới thiệu trong môn học, Sinh viên cần tìm hiểu thêm trên mạng, các bài báo, diễn đàn để tổng hợp thêm các khái niệm liên quan lập trình hướng đối tượng.

Câu 53: Phân biệt Overloading và Overriding Method**Yêu cầu:**

Sinh viên hãy phân biệt Overloading và Overriding Method, cho ví dụ minh họa.

Hướng dẫn:

Sinh viên cần tạo bảng so sánh để làm rõ các điểm khác biệt và ý nghĩa của Overloading, Overriding Method.

Câu 54: Giải thích chi tiết Alias và cơ chế gom rác tự động**Yêu cầu:**

Sinh viên hãy chi tiết Alias và cơ chế gom rác tự động, cho ví dụ minh họa và giải thích.

Hướng dẫn:

Từ các kiến thức được giới thiệu trong môn học, Sinh viên cần tìm hiểu thêm trên mạng, các bài báo, diễn đàn để trả lời câu hỏi.

Câu 55: Trình bày các kỹ thuật khai báo phương thức tĩnh và ý nghĩa**Yêu cầu:**

Sinh viên hãy trình bày các kỹ thuật khai báo phương thức tĩnh và ý nghĩa sử dụng của phương thức tĩnh.

Hướng dẫn:

Từ các kiến thức được giới thiệu trong môn học, Sinh viên cần tìm hiểu thêm trên mạng, các bài báo, diễn đàn để trả lời câu hỏi.

Câu 56: Trình bày các kỹ thuật kế thừa và ý nghĩa

Yêu cầu:

Sinh viên hãy trình bày các kỹ thuật kế thừa và ý nghĩa của việc sử dụng kế thừa, cho ví dụ minh họa.

Hướng dẫn:

Từ các kiến thức được giới thiệu trong môn học, Sinh viên cần tìm hiểu thêm trên mạng, các bài báo, diễn đàn để trả lời câu hỏi.

Câu 57: Trình bày kỹ thuật Singleton và ý nghĩa sử dụng

Yêu cầu:

Sinh viên hãy trình bày kỹ thuật Singleton và ý nghĩa sử dụng, cho ví dụ minh họa

Hướng dẫn:

Từ các kiến thức được giới thiệu trong môn học, Sinh viên cần tìm hiểu thêm trên mạng, các bài báo, diễn đàn để trả lời câu hỏi.

Câu 58: Xây dựng lớp Nhân viên

Yêu cầu:

Sinh viên hãy xây dựng lớp NhanVien theo mô tả như dưới đây:

- Họ, Tên, Số sản phẩm lần lượt là các thuộc tính họ, tên và số sản phẩm của nhân viên.
- Viết hàm khởi tạo cho NhanVien, hàm này sẽ khởi tạo họ, tên, số sản phẩm của nhân viên; hàm phải kiểm tra số sản phẩm là số lớn hơn hoặc bằng 0, nếu là số âm thì gán giá trị cho số sản phẩm bằng 0.

- Viết các hàm lấy và gán giá trị cho thuộc tính của lớp (các hàm getter/setter).
- Viết hàm getLuong() để tính lương cho nhân viên, lương = số sản phẩm * đơn giá, với đơn giá tùy thuộc vào số sản phẩm như sau:

Số sản phẩm	Đơn giá
1 - 199	0.5
200 - 399	0.55
400 - 599	0.6
600 trở lên	0.65

- Viết hàm IsHigher(NhanVien nv2): Hàm này trả về giá trị true khi số sản phẩm (SoSP) của đối tượng hiện tại lớn hơn số sản phẩm của nv2, ngược lại trả về false

- Viết các mã lệnh để sử dụng lớp NhanVien theo yêu cầu sau:
 - + Cho người dùng nhập vào 2 nhân viên, mỗi nhân viên nhập vào họ, tên, số sản phẩm của họ. Hãy tính và xuất ra lương của từng nhân viên. So sánh và xuất ra thông báo nhân viên nào có số sản phẩm nhiều hơn và nhiều hơn bao nhiêu. Dùng 2 cách so sánh: dùng hàm IsHigher và không dùng hàm IsHigher.
 - + Tạo các menu để cho người dùng nhập Nhân viên và sử dụng các hàm liên quan tới Nhân viên.

Hướng dẫn:

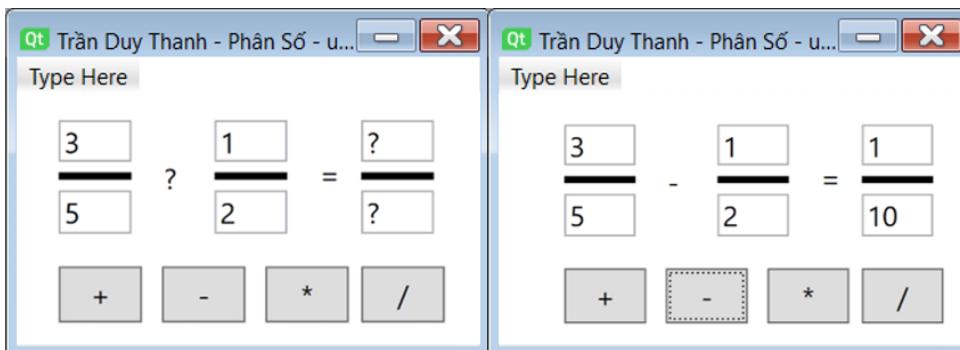
Từ các kiến thức được giới thiệu trong môn học, Sinh viên cần tìm hiểu thêm trên mạng, các bài báo, diễn đàn để trả lời câu hỏi.

Câu 59: Xây dựng lớp Phân Số

Yêu cầu:

Sinh viên hãy xây dựng lớp Phân Số, mỗi một Phân số có tử số và mẫu số cùng với các phương thức: Cộng, Trừ, Nhân Chia, Tối Giản.

Viết các mã lệnh để thực nghiệm các đối tượng phân số và các phương thức tương ứng. Sau đó sinh viên hãy thiết kế giao diện tương tác như dưới đây:



Hướng dẫn:

Từ các kiến thức được giới thiệu trong môn học, Sinh viên cần tìm hiểu thêm trên mạng, các bài báo, diễn đàn để trả lời câu hỏi.

Câu 60: Phần mềm quản lý Sinh viên

Yêu cầu:

Cho cấu trúc của lớp Sinh viên có mô tả như dưới đây:

- Thuộc tính:
 - + Mã số sinh viên (kiểu chuỗi)
 - + Họ và tên (kiểu chuỗi)
 - + Ngày tháng năm sinh (dùng Datetime)
 - + Họ (kiểu chuỗi, chỉ đọc, lấy từ Họ và tên)
 - + Tên (kiểu chuỗi, chỉ đọc, lấy từ Họ và tên)
 - + Tuổi (kiểu int, chỉ đọc, tính từ ngày tháng năm sinh)
- Phương thức
 - + In thông tin: trả về chuỗi thông tin sinh viên
 - Tạo thêm một Lớp có sử dụng cấu trúc dữ liệu phù hợp để quản lý danh sách Sinh viên và cung cấp các phương thức xử lý sau:
 - + Tính tổng số sinh viên có trong danh sách
 - + Tìm sinh viên theo họ tên
 - + Tìm sinh viên có ngày sinh trong tháng hiện tại

+ Sắp xếp mảng sinh viên tăng dần theo tuổi

+ Nâng cao (*): Thiết kế giao diện để sử dụng các chức năng của chương trình

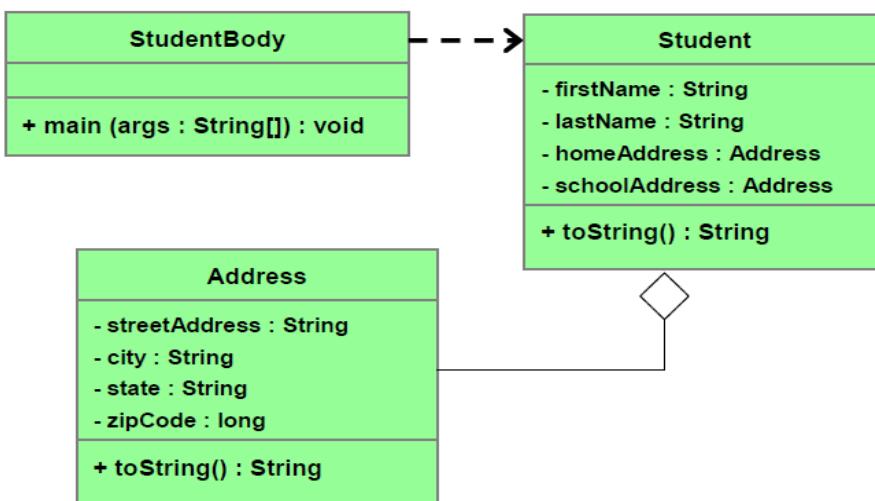
Hướng dẫn:

Từ các kiến thức được giới thiệu trong môn học, Sinh viên cần tìm hiểu thêm trên mạng, các bài báo, diễn đàn để trả lời câu hỏi.

Câu 61: Phần mềm quản lý Sinh viên

Yêu cầu:

Cho mô hình lớp có mối quan hệ như dưới đây:



Hãy thiết kế lớp Student và lớp Address, khai báo các thuộc tính và sử dụng kiểu dữ liệu phù hợp. Bổ sung tham chiếu đối tượng cho Student để quản lý các Address.

Tạo file StudentBody để thử nghiệm các chức năng của chương trình. Yêu cầu phần mềm cần cung cấp các chức năng:

+ Xem, thêm, sửa, xóa Student

+ Quản lý danh sách Student

+ Lọc Student theo địa chỉ (Sinh viên tùy chọn loại địa chỉ)

Hướng dẫn:

Từ các kiến thức được giới thiệu trong môn học, Sinh viên cần tìm hiểu thêm trên mạng, các bài báo, diễn đàn để trả lời câu hỏi.

Câu 62: Phần mềm Quản lý Sản phẩm

Yêu cầu:

Viết chương trình quản lý bán hàng được mô tả như bên dưới, yêu cầu dùng List hoặc Dictionary để quản lý danh sách :

- Một Danh mục có nhiều sản phẩm, thông tin danh mục: Mã, tên danh mục.
- Thông tin sản phẩm gồm: Mã, tên sản phẩm, giá cả, xuất xứ.
- Hãy thiết kế các mô hình lớp phù hợp với mô tả
- Viết hàm cho phép nhập thông tin Danh mục và Sản phẩm
- Xuất danh sách sản phẩm thuộc một danh mục bất kỳ
- Cho phép cập nhật thông tin sản phẩm
- Xóa sản phẩm bất kỳ trong Danh mục
- Thống kê tổng giá trị của các mặt hàng
- Liệt kê các sản phẩm có xuất xứ từ Trung quốc

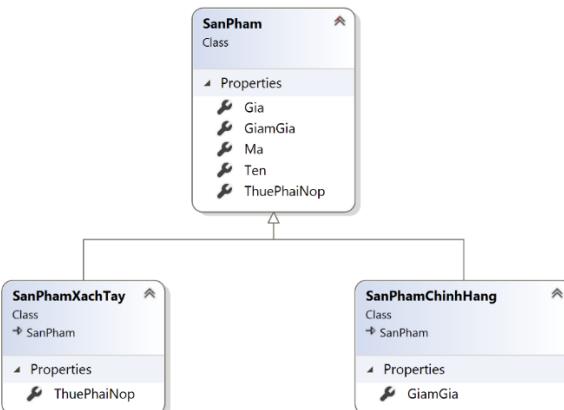
Hướng dẫn:

Từ các kiến thức được giới thiệu trong môn học, Sinh viên cần tìm hiểu thêm trên mạng, các bài báo, diễn đàn để trả lời câu hỏi.

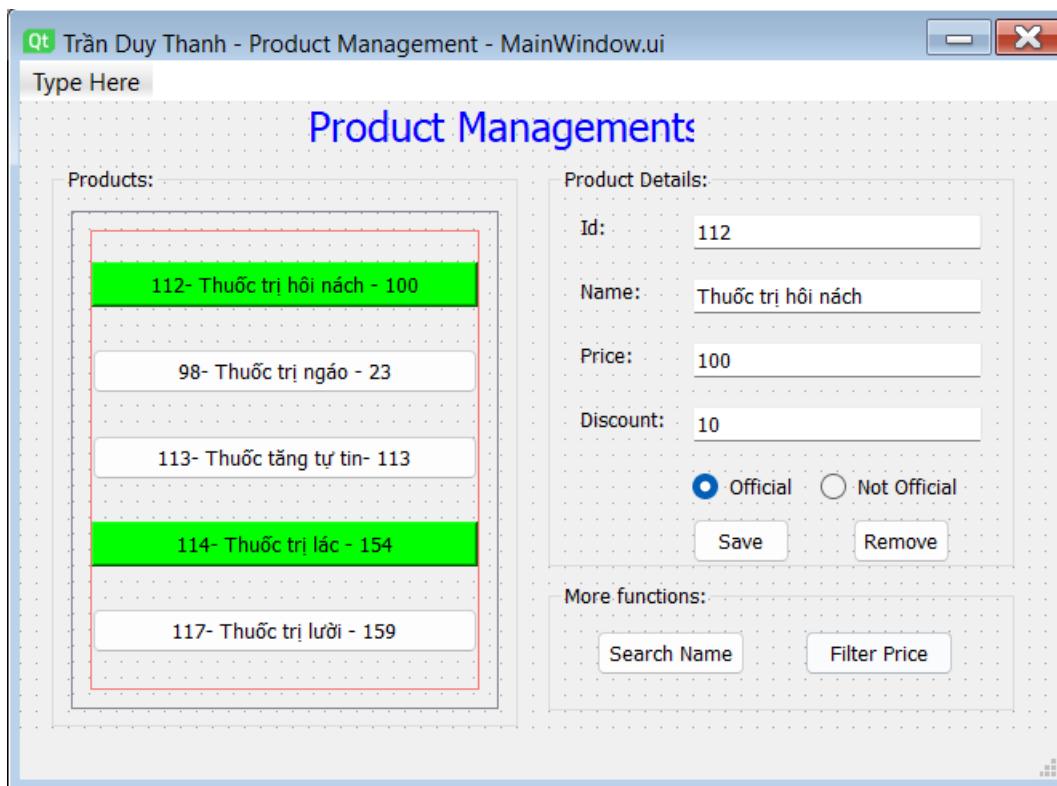
Câu 63: Phần mềm quản lý Sản phẩm – kế thừa

Yêu cầu:

Cho mô lớp Sản phẩm có dạng kế thừa như dưới đây:



- Thông tin của một sản phẩm gồm: Mã (kiểu chuỗi), Tên (kiểu chuỗi), Giá (kiểu double), Giảm Giá (double) và Thuế phải nộp (double).
- Có hai loại Sản phẩm: Sản phẩm xách tay và Sản phẩm chính hãng cùng kê thùng từ lớp Sản phẩm. Hàng xách tay thì đóng thuế 0% và được giảm giá 8%, còn hàng Chính hãng phải đóng thuế 10 % nhưng không giảm giá.
- Thiết kế các lớp phù hợp theo mô tả ở trên, sử dụng cấu trúc dữ liệu phù hợp để lưu trữ danh sách các sản phẩm, viết các mã lệnh để thực nghiệm chương trình.
- Nâng cao (*) thiết kế thêm giao diện tương tác:



- + Các sản phẩm sẽ được nạp lên giao diện khi nhấn nút “Save”, nếu sản phẩm là chính hãng thì nền của QPushButton tô màu xanh.
- + Nút “Save” vừa cung cấp tính năng thêm mới và cập nhật(nếu id đã tồn tại)
- + Nút “Remove” xóa Product đang chọn trên giao diện
- + Các nút như “Search Name”, “Filter Price”. Sinh viên tự đề xuất giao diện phù hợp để thực hiện các chức năng này.

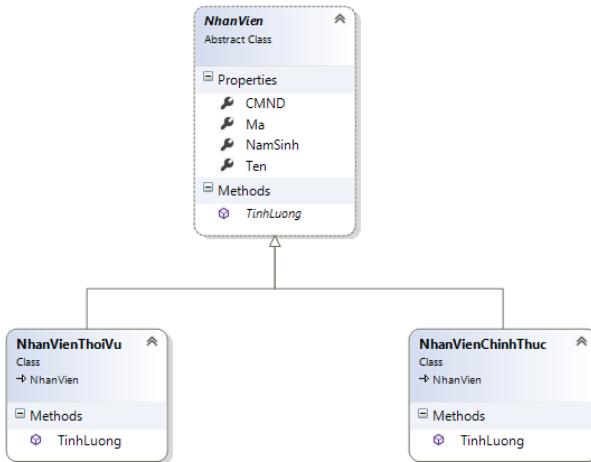
Hướng dẫn:

Từ các kiến thức được giới thiệu trong môn học, Sinh viên cần tìm hiểu thêm trên mạng, các bài báo, diễn đàn để trả lời câu hỏi.

Câu 64: Phần mềm quản lý Nhân viên – Kế thừa

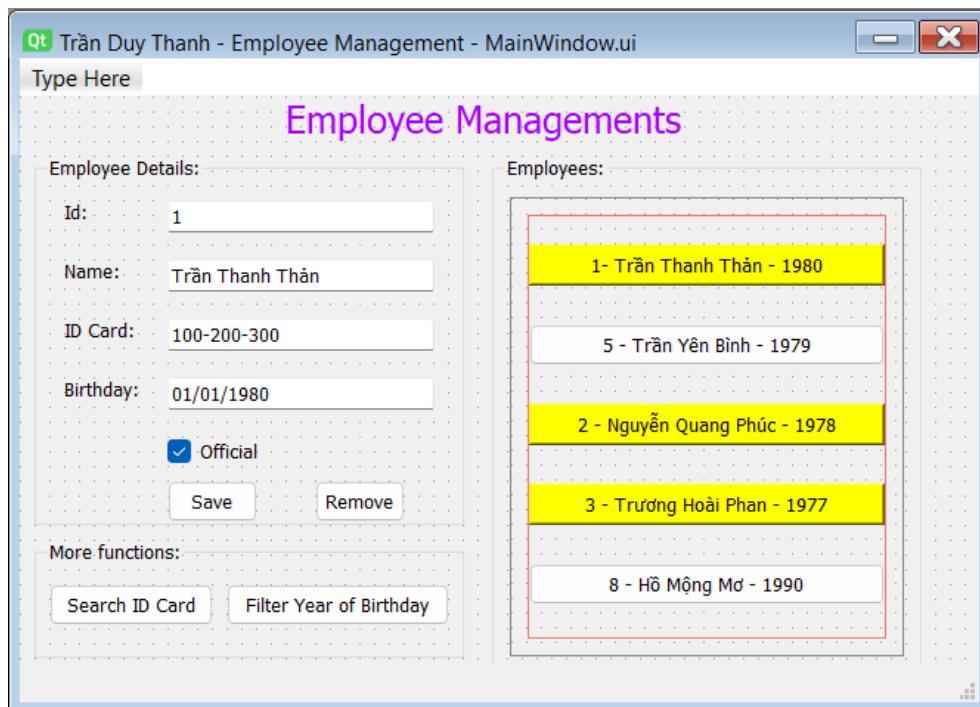
Yêu cầu:

Cho mô hình lớp Nhân viên có dạng kế thừa như dưới đây:



Mặc định nhân viên thời vụ được tính 1 ngày 300 nghìn, còn nhân viên chính thức có lương hàng tháng là 20 triệu.

Giao diện tương tác như sau:



-Hãy thiết kế các lớp phù hợp để đáp ứng mô tả ở trên, sử dụng cấu trúc dữ liệu phù hợp để lưu trữ danh sách các Nhân viên, viết các mã lệnh để thực nghiệm chương trình:

- + Các thao tác: Xem, thêm, sửa, xóa nhân viên
- + Nhân viên chính thức tô nền vàng
- + Sinh viên đề xuất thực hiện các chức năng Search ID Card, Filter Year.

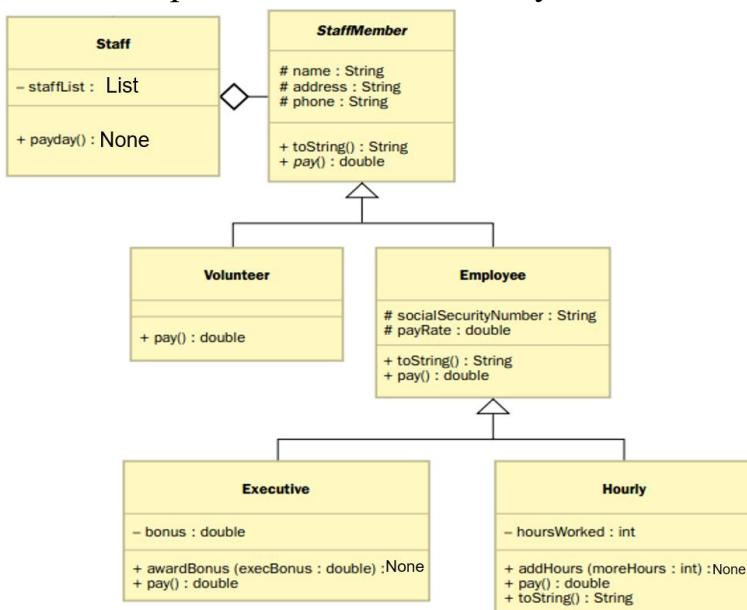
Hướng dẫn:

Từ các kiến thức được giới thiệu trong môn học, Sinh viên cần tìm hiểu thêm trên mạng, các bài báo, diễn đàn để trả lời câu hỏi.

Câu 65: Phần mềm quản lý Nhân viên – Kế thừa(*)

Yêu cầu:

Cho mô hình kế thừa các lớp nhân viên như dưới đây:



Sinh viên hãy tự thiết kế các lớp đối tượng cùng các thuộc tính phù hợp như hình mô tả, cung cấp thêm các phương thức để xử lý dữ liệu các đối tượng. Bổ sung mã lệnh để thực nghiệm chương trình.

Hướng dẫn:

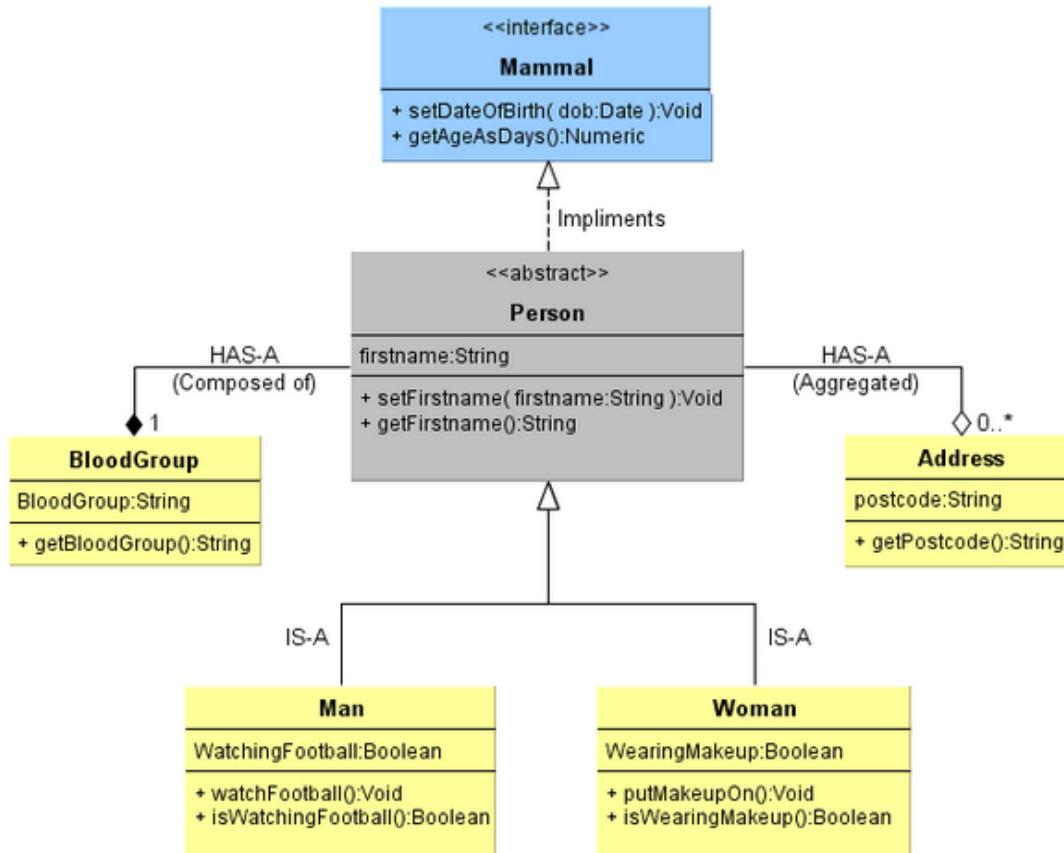
Từ các kiến thức được giới thiệu trong môn học, Sinh viên cần tìm hiểu thêm trên mạng, các bài báo, diễn đàn để trả lời câu hỏi.

- Hàm `pay()` của **Volunteer** là 100, của **Employee** là 3000
- Hàm `pay()` của **Executive** là `pay()` của **Employee** +`awardBonus`
- Hàm `pay()` của **Hourly** = $10 * \text{hoursWorked}$
- Lớp **Staff** là lớp quản lý toàn bộ **StaffMember**

Câu 66: Phần mềm quản lý Dân số (*)

Yêu cầu:

Cho các mô hình lớp kế thừa như dưới đây:



Hãy thiết kế các Lớp cùng với các thuộc tính và phương thức như mô tả.

Bổ sung thêm các collection phù hợp để lưu trữ mối quan hệ giữa các đối tượng.

Viết mã lệnh để thử nghiệm các chức năng của chương trình.

Hướng dẫn:

Từ các kiến thức được giới thiệu trong môn học, Sinh viên cần tìm hiểu thêm trên mạng, các bài báo, diễn đàn để trả lời câu hỏi.

Câu 67: Phần mềm quản lý Đối tượng toán học

Yêu cầu:

Viết chương trình hướng đối tượng dùng để tính chu vi và diện tích hình tròn, hình chữ nhật và hình vuông. Với yêu cầu như sau:

Xây dựng lớp **Hinh** có:

-Các thuộc tính:

+mau: màu sắc của hình có kiểu dữ liệu là String.

-Các phương thức:

+Hinh(String mau): phương thức khởi tạo đối tượng hình.

+TinhDienTich(): phương thức tính diện tích hình.

+TinhChuVi(): phương thức tính chu vi hình.

+LayThongTin(): phương thức trả về các thông tin của đối tượng hình được tạo kiểu String.

-Xây dựng lớp **HinhChuNhat** kế thừa lớp **Hinh** và có thêm các thuộc tính:

+chieudai: chiều dài của hình chữ nhật có kiểu dữ liệu là double.

+chieurong: chiều rộng của hình chữ nhật có kiểu dữ liệu là double.

-Xây dựng lớp **HinhTron** kế thừa từ lớp **Hinh** và có thêm thuộc tính:

+bankinh: bán kính của hình tròn có kiểu dữ liệu là double.

-Xây dựng lớp **HinhVuong** kế thừa từ lớp **HinhChuNhat**

-Xây dựng lớp **KiemTraHinh** với các yêu cầu sau:

+Tạo 3 đối tượng hinhChuNhat, hinhTron, hinhVuong từ 3 constructor HinhChuNhat(mau, chieudai, chieurong), HinhTron(mau, bankinh), HinhVuong(mau, bankinh).

+Tính diện tích và chu vi của các đối tượng hinhChuNhat, hinhTron, hinhVuong đã tạo.

+Hiển thị thông tin các đối tượng Hinh đã tạo ra màn hình console.

-Chạy chương trình và quan sát kết quả.

Hướng dẫn:

Từ các kiến thức được giới thiệu trong môn học, Sinh viên cần tìm hiểu thêm trên mạng, các bài báo, diễn đàn để trả lời câu hỏi.

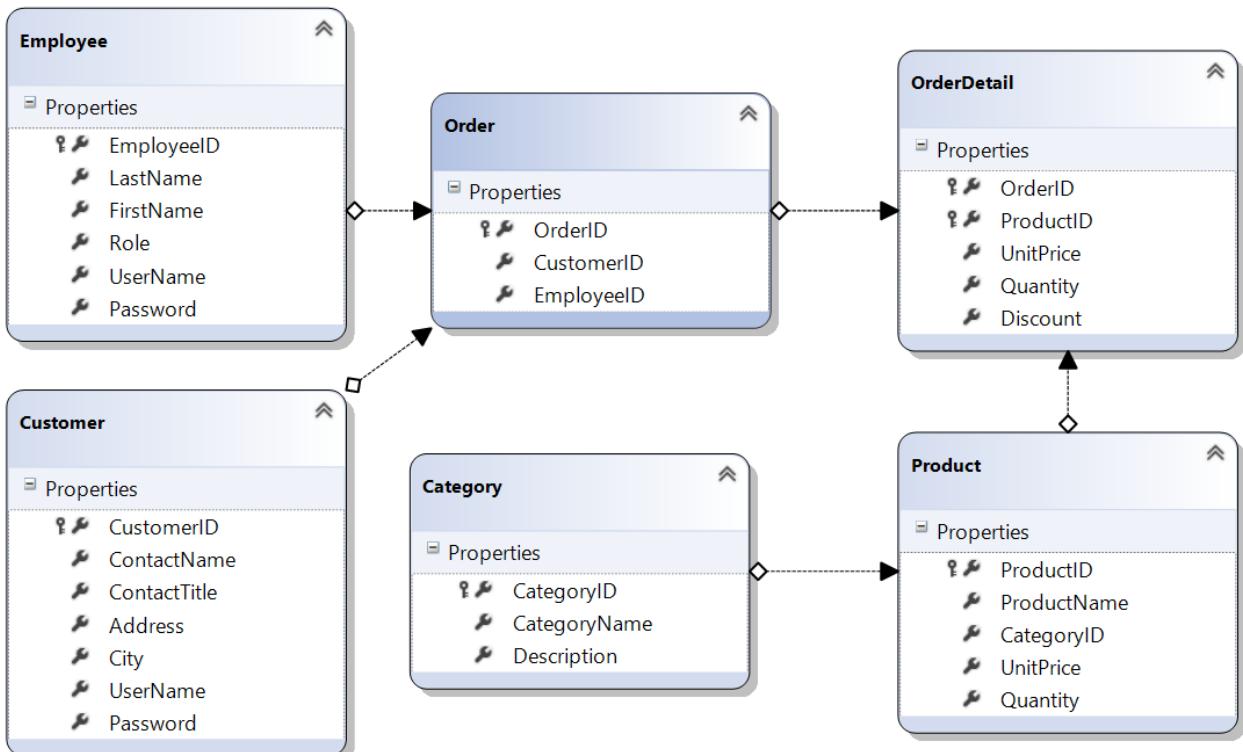
Sinh viên tạo các lớp đối tượng riêng biệt ở các file python độc lập

Các lớp kế thừa cần biết tái sử dụng các hàm và thuộc tính.

Câu 68: Phần mềm quản lý Bán hàng (*)

Yêu cầu:

Cho một phần mô hình class trong hệ thống Quản lý bán hàng của Sàn thương mại điện tử, các đối tượng có dạng mối quan hệ như dưới đây:



-Với các thuộc tính có ID đều là kiểu số nguyên

-Quantity là số nguyên

-UnitPrice và Discount là số thực

-Tất cả các thuộc tính còn lại là kiểu chuỗi

Về mối quan hệ giữa các đối tượng được mô tả như sau:

+Một danh mục sản phẩm sẽ có nhiều sản phẩm, và một sản phẩm chỉ thuộc về một danh mục

+Một hóa đơn cho biết khách hàng nào mua, nhân viên nào bán

+Một chi tiết hóa đơn: Cho biết chi tiết từng sản phẩm được mua trong từng hóa đơn cụ thể

Sinh viên hãy thiết kế 6 lớp đối tượng ở trên, khai báo các thuộc tính và kiểu dữ liệu phù hợp, sử dụng List để quản lý mối quan hệ này.

Sau khi thiết kế xong các mô hình lớp, Sinh viên hãy viết các mã lệnh để tạo mẫu dữ liệu như sau:

- + 3 Danh mục
- + 10 Sản phẩm được phân bổ vào các danh mục
- + 3 Nhân viên
- + 5 Khách hàng
- + 10 Hóa đơn
- + Mỗi hóa đơn có từ 3 tới 5 chi tiết hóa đơn

Cung cấp thêm các mã lệnh để xem chi tiết các loại dữ liệu trên (mỗi đối tượng đều có danh sách dữ liệu và mối quan hệ liên quan).

Hướng dẫn:

Từ các kiến thức được giới thiệu trong môn học, Sinh viên cần tìm hiểu thêm trên mạng, các bài báo, diễn đàn để trả lời câu hỏi.

Câu 69: Phần mềm quản lý thư viện (*)

Yêu cầu:

Thiết kế các mô hình lớp đối tượng cho một hệ thống quản lý thư viện và cung cấp các phương thức cho phép thêm/xoá/sửa thông tin sách và thành viên, quản lý việc mượn/trả sách và tính phí phạt khi trễ hạn.

Đặc điểm:

Các lớp: Sách, Thành viên, Phiếu mượn sách.

Các phương thức: Tìm kiếm sách theo nhiều tiêu chí, tính toán phí trễ hạn, đăng ký thành viên mới, lập phiếu mượn sách,...

Sinh viên tự suy luận các thuộc tính phù hợp với từng lớp đối tượng để khai báo.

Hướng dẫn:

Từ các kiến thức được giới thiệu trong môn học, Sinh viên cần tìm hiểu thêm trên mạng, các bài báo, diễn đàn để trả lời câu hỏi.

Sinh viên hãy đổi chiếu với thư viện của Trường mình đang học để thực hiện.

Câu 70: Phần mềm quản lý đặt phòng khách sạn (*)

Yêu cầu:

Thiết kế các mô hình lớp đối tượng cho một hệ thống quản lý đặt phòng khách sạn với các chức năng đặt phòng, hủy đặt phòng, tính tiền thanh toán,...

Đặc điểm:

Các lớp: Phòng, Khách hàng, Đặt phòng,...

Các phương thức: Đặt phòng, hủy phòng, tính tiền, in hóa đơn,...

Sinh viên tham khảo các hệ thống đặt khách sạn trên mạng để đề xuất các lớp cũng như các thuộc tính phù hợp với từng lớp đối tượng để khai báo.

Viết mã lệnh để thực nghiệm chương trình.

Hướng dẫn:

Từ các kiến thức được giới thiệu trong môn học, Sinh viên cần tìm hiểu thêm trên mạng, các bài báo, diễn đàn để trả lời câu hỏi.

Câu 71: Phần mềm mô phỏng hệ thống ATM (*)

Yêu cầu:

Thiết kế các mô hình lớp đối tượng đơn giản cho một hệ thống hệ thống ATM với các chức năng rút tiền, kiểm tra số dư, chuyển khoản,... mô tả sơ lược:

Các lớp: Ngân hàng, Tài khoản, Giao dịch,...

Các phương thức: Rút tiền, chuyển khoản, kiểm tra số dư, lịch sử giao dịch,...

Sinh viên tham khảo các hệ thống ATM thường ngày để đề xuất các lớp cũng như các thuộc tính phù hợp với từng lớp đối tượng để khai báo.

Viết mã lệnh để thực nghiệm chương trình.

Hướng dẫn:

Từ các kiến thức được giới thiệu trong môn học, Sinh viên cần tìm hiểu thêm trên mạng, các bài báo, diễn đàn để trả lời câu hỏi.

Câu 72: Phần mềm mô phỏng hệ thống đặt vé xe bus/tàu hỏa (*)**Yêu cầu:**

Thiết kế các mô hình lớp đối tượng đơn giản cho một hệ thống đặt vé xe bus hoặc tàu hỏa với các chức năng đặt vé, hủy vé, kiểm tra lịch trình,...:

Các lớp: Chuyến đi, Vé, Hành khách,...

Các phương thức: Đặt vé, hủy vé, tính tổng doanh thu, in vé,...

Sinh viên tham khảo các hệ thống đặt vé xe của các hãng ở Việt Nam để đề xuất các lớp cũng như các thuộc tính phù hợp với từng lớp đối tượng để khai báo.

Viết mã lệnh để thực nghiệm chương trình.

Hướng dẫn:

Từ các kiến thức được giới thiệu trong môn học, Sinh viên cần tìm hiểu thêm trên mạng, các bài báo, diễn đàn để trả lời câu hỏi.

Câu 73: Phần mềm mô phỏng hệ thống quản lý bệnh viện (*)**Yêu cầu:**

Thiết kế mô hình lớp một hệ thống quản lý bệnh viện với các yêu cầu sau:

-Tạo lớp **Patient** với các thuộc tính như name, age, illness, doctor_assigned.

-Tạo lớp **Doctor** với các thuộc tính như name, specialty, patients_assigned.

-Tạo lớp **Hospital** để quản lý danh sách Patient và Doctor, bao gồm các phương thức như thêm/xóa bệnh nhân, thêm/xóa bác sĩ, gán bác sĩ cho bệnh nhân, v.v.

Viết chương trình sử dụng các lớp trên để thực hiện các chức năng quản lý bệnh viện.

Hướng dẫn:

Từ các kiến thức được giới thiệu trong môn học, Sinh viên cần tìm hiểu thêm trên mạng, các bài báo, diễn đàn để trả lời câu hỏi.

MODULE 4

THAO TÁC VỚI TẬP TIN

Nội dung kiến thức thực hành:

- + Xử lý tập tin văn bản .txt
- + Xử lý tập tin cấu trúc XML
- + Xử lý tập tin cấu trúc JSON.
- + Xử lý tập tin cấu trúc excel (csv, xls, xlsx).
- + Xử lý Pickle File
- + Xử lý tập tin đa phương tiện
- + Mapping tương tác đối tượng với các tập tin

Câu 74: Trình bày kỹ thuật xử lý tập tin .txt

Yêu cầu:

Trình bày các kỹ thuật xử lý tập tin .txt:

- Các lớp, hàm được sử dụng
- Lưu text file có định dạng unicode
- Cách đọc text file có định dạng unicode

Hướng dẫn:

Từ các kiến thức được giới thiệu trong môn học, Sinh viên cần tìm hiểu thêm trên mạng, các bài báo, diễn đàn để trả lời câu hỏi.

Câu 75: Trình bày kỹ thuật xử lý tập tin XML

Yêu cầu:

Trình bày các kỹ thuật xử lý tập tin XML:

- Các lớp, hàm được sử dụng
- Lưu XML file
- Cách đọc XML file
- Cách phân tích để đọc dữ liệu trong các Element, Attribute

Hướng dẫn:

Từ các kiến thức được giới thiệu trong môn học, Sinh viên cần tìm hiểu thêm trên mạng, các bài báo, diễn đàn để trả lời câu hỏi.

Câu 76: Trình bày kỹ thuật xử lý tập tin JSON

Yêu cầu:

Phân biệt JSON Object và JSON Array, Trình bày các kỹ thuật xử lý tập tin JSON:

- Các lớp, hàm được sử dụng
- Cách lưu JSON
- Cách đọc JSON
- Cách phân tích để đọc dữ liệu trong các đối tượng JSON Object và JSON Array.
- Cách xử lý dữ liệu có Unicode
- Cách đọc dữ liệu từ web API
- Cách chuyển đổi dữ liệu qua lại giữa XML và JSON

Hướng dẫn:

Từ các kiến thức được giới thiệu trong môn học, Sinh viên cần tìm hiểu thêm trên mạng, các bài báo, diễn đàn để trả lời câu hỏi.

Câu 77: Trình bày kỹ thuật xử lý tập tin Excel (CSV, XLS, XLSX)

Yêu cầu:

Trình bày các kỹ thuật xử lý tập tin CSV, XSL, XLSX:

- Các lớp, hàm được sử dụng tương ứng với CSV, XLS, XLSX

- Cách lưu dữ liệu với CSV và XLS/XLSX file
- Cách đọc dữ liệu với CSV và XLS/XLSX file
- Cách phân tích để đọc dữ liệu trong các thành phần của XLS/XLSX

Hướng dẫn:

Từ các kiến thức được giới thiệu trong môn học, Sinh viên cần tìm hiểu thêm trên mạng, các bài báo, diễn đàn để trả lời câu hỏi.

Câu 78: Trình bày kỹ thuật xử lý Pickle File

Yêu cầu:

Sinh viên trình bày các kỹ thuật xử lý Pickle File:

- Mô tả cách tạo lớp mô hình để tương tác với Pickle File
- Chức năng lưu mô hình lớp đối tượng vào ổ cứng
- Chức năng tải dữ liệu từ ổ cứng vào bộ nhớ và khôi phục đối tượng
- Sử dụng Pickle cho lớp Sản phẩm với các thuộc tính: ID, tên, số lượng, đơn giá

Hướng dẫn:

Sinh viên xem lại kiến thức được trình bày trong môn học và tìm hiểu thêm trên mạng để thực hiện bài làm.

Câu 79: Xử lý số trong Text File

Yêu cầu:

Cho một tập tin có dữ liệu trên mỗi dòng như dưới đây:

5,6,8,9,-5
-9,5,4,7,8
6,7,8,3,6,46,7,2,-6,-7

- (a) Viết hàm đọc file, mỗi dòng khởi tạo thành 1 list và xuất ra màn hình
- (b) Xuất các số âm trên mỗi dòng ra màn hình
- (c) Với mỗi dòng trên màn hình, kèm theo xuất số nhỏ nhất
- (d) Với mỗi dòng trên màn hình, kèm theo xuất số lớn nhất

(e) Với mỗi dòng trên màn hình: Dòng lẻ các số cần được sắp xếp tăng dần, dòng chẵn các số cần được sắp xếp giảm dần

Hướng dẫn:

Sinh viên xem lại kiến thức được trình bày trong môn học và tìm hiểu thêm trên mạng để thực hiện bài làm.

Dưới đây là một số hướng dẫn mẫu, Sinh viên có thể sử dụng và mở rộng cho các câu còn lại:

Bước 1: Tạo file “FileProcessing.py” có mã lệnh:

```
1 def WriteData(path,data):
2     file=open(path, 'a', encoding='utf-8')
3     file.writelines(data)
4     file.writelines("\n")
5     file.close()
6 def ReadData(path):
7     arrNumber=[]
8     file=open(path, 'r', encoding='utf-8')
9     for line in file:
10         data=line.strip()
11         arr=data.split(',')
12         arrNumber.append(arr)
13     file.close()
14     return arrNumber
```

Bước 2: Tạo file “TestWriteFile.py” có mã lệnh:

```
1 from Ex79.FileProcessing import WriteData
2
3 WriteData("database_number.txt", "-5,4,7,9,3,20")
4 WriteData("database_number.txt", "5,-4,37,-19,24,-21")
5 WriteData("database_number.txt", "15,9,0,-38,-3,15")
6 WriteData("database_number.txt", "5,-4,77,-9,3,-7")
7 WriteData("database_number.txt", "55,44,27")
8 WriteData("database_number.txt", "-50,26")
```

Bước 3: Tạo file “TestReadFile.py” có mã lệnh:

```
1 from FileProcessing import *
2 arrNumber=ReadData("database_number.txt")
3 print(arrNumber)
4 def printNegativeNumberPerLine(arrNumber):
5     for row in arrNumber:
6         for element in row:
```

```
7     number=int(element)
8     if number<0:
9         print(number,end='\t')
10    print()
11 print("Negative Number per Line:")
12 printNegativeNumberPerLine(arrNumber)
```

Khi test thì chạy mã lệnh lưu file trước, sau đó chạy đọc file. Chúng ta sẽ có kết quả như mong muốn.

Câu 80: Chương trình quản lý Sản phẩm – Text File

Yêu cầu:

Viết chương trình nhập vào thông tin của một sản phẩm bao gồm các thuộc tính:

- Mã sản phẩm: Kiểu chuỗi
- Tên sản phẩm: Kiểu chuỗi
- Đơn Giá: Kiểu số

Mỗi một Sản phẩm sau khi nhập thành công sẽ lưu nối đuôi vào File theo quy tắc:

MSSP;Tên Sản phẩm;Đơn giá

Mẫu Dữ liệu lưu nối đuôi vào file tương tự như dưới đây:

```
sv1;Cocacolala;15.5
sp2;Bưởi Thanh Thản;18.0
sp3;Bia 333;14.5
```

Ngoài ra chương trình cần thực hiện các chức năng chính:

- (a) Cần thiết kế hướng đối tượng
- (b) Cung cấp các chức năng: Xem, thêm, sửa, xóa sản phẩm
- (c) Mọi thay đổi dữ liệu đều tự động lưu xuống tập tin: Từ danh sách đối tượng sản phẩm, trích xuất dữ liệu để lưu xuống tập tin đúng định dạng
- (d) Đọc danh sách sản phẩm từ File đã được lưu: Từ dữ liệu đọc được, mô hình hóa lại hướng đối tượng để quản lý

- (e) Sắp xếp Sản phẩm theo đơn giá giảm dần
- (f) Sắp xếp Sản phẩm theo đơn giá tăng dần
- (g*) Bổ sung giao diện tương tác người dùng PyQt6 – Qt Designer

Hướng dẫn:

Sinh viên xem lại kiến thức được trình bày trong môn học và tìm hiểu thêm trên mạng để thực hiện bài làm.

Dưới đây là một số mã lệnh mẫu được cung cấp, Sinh viên có thể dựa vào minh họa này để thực hiện các câu còn lại:

Bước 1: Khai báo lớp đối tượng Product trong file Product.py:

```
1 class Product:  
2     def __init__(self,productId,productName,unitPrice):  
3         self.productId=productId  
4         self.productName=productName  
5         self.unitPrice=unitPrice  
6     def __str__(self):  
7         return f'{self.productId}\t{self.productName}\t{self.unitPrice}'
```

Bước 2: Khai báo lớp đối tượng FileFactory trong file FileFactory.py

```
1 import traceback  
2  
3 from Ex80.Product import Product  
4  
5  
6 class FileFactory:  
7     @staticmethod  
8     def writeData(path,product):  
9         try:  
10             line = (product.productId + ";" +  
11                     product.productName + ";" +  
12                     str(product.unitPrice))  
13             file = open(path, 'a', encoding='utf-8')  
14             file.writelines(line)  
15             file.writelines("\n")  
16             file.close()  
17             return True  
18         except:  
19             traceback.print_exc()  
20             return False  
21     @staticmethod
```

```

22 def readData(path):
23     products = []
24     try:
25         file = open(path, 'r', encoding='utf-8')
26         for line in file:
27             data = line.strip()
28             arr = data.split(';')
29             product=Product(arr[0],arr[1],float(arr[2]))
30             products.append(product)
31         file.close()
32     return products
33     pass
34 except:
35     traceback.print_exc()
36 return []

```

Bước 3: Khai báo thử nghiệm chức năng lưu file TestWriteData.py có mapping đối tượng

```

1 from Ex80.Factory import FileFactory
2 from Ex80.Product import Product
3 print("Input Products:")
4 while True:
5     productId=input("Input Product ID:")
6     productName=input("Input Product Name:")
7     unitPrice=float(input("Input Unit Price:"))
8     product=Product(productId,productName,unitPrice)
9
10    FileFactory.writeData("database.txt",product)
11    ans=input("Continue?(Y/N):")
12    if ans!='Y' or ans!='y':
13        break

```

Bước 4: Khai báo thử nghiệm chức năng đọc file TestReadData.py có mapping đối tượng

```

1 from Ex80.Factory import FileFactory
2
3 products=FileFactory.readData("database.txt")
4 def printProducts(products):
5     for product in products:
6         print(product)
7     print()
8

```

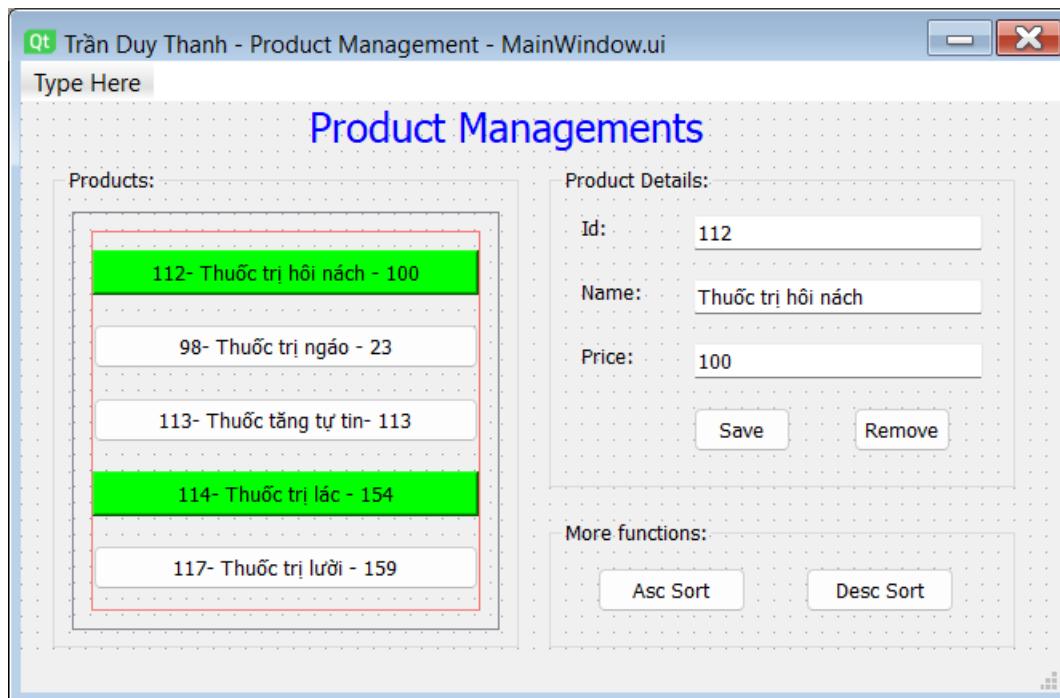
```

9 printProducts(products)
10 def sortProducts(products):
11     for i in range(len(products)):
12         for j in range(len(products)):
13             pi=products[i]
14             pj=products[j]
15             if pi.unitPrice<pj.unitPrice:
16                 products[i]=pj
17                 products[j]=pi
18 sortProducts(products)
19 print("Products after UnitPrice Sorting:")
20 printProducts(products)

```

Khi thực nghiệm, ta chạy bước 3 trước để nhập dữ liệu, sau đó chạy bước 4 sẽ có kết quả như mong muốn.

Đối với câu (g*) Sinh viên có thể thiết kế giao diện như dưới đây, khi chạy phần mềm nếu có dữ liệu trong file thì tự động nạp lên giao diện. Mọi thao tác của người dùng chương trình sẽ lưu dữ liệu xuống text file:



- (g1) Khởi động phần mềm, nếu có file dữ liệu, chương trình sẽ nạp đối tượng Product lên giao diện phần “Products”
- (g2) Thao tác Save, Remove chương trình sẽ lưu đè lại dữ liệu xuống file
- (g3) Asc Sort: Sắp xếp sản phẩm theo giá tăng dần, nạp lại giao diện
- (g4) Desc Sort: Sắp xếp sản phẩm theo giá giảm dần, nạp lại giao diện

Câu 81: Chương trình quản lý Employee – XML

Yêu cầu:

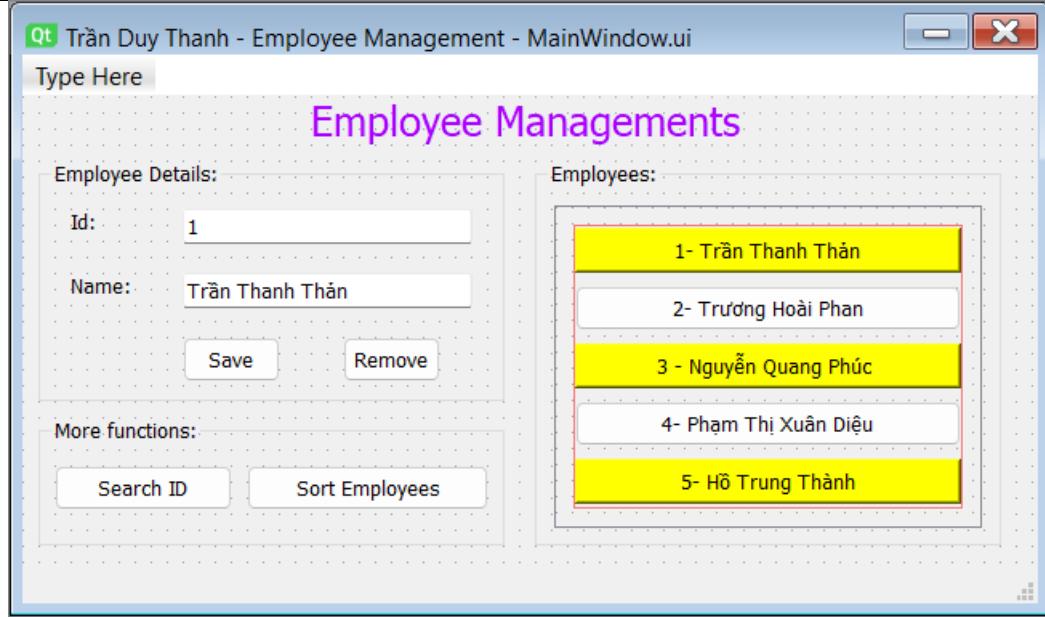
Cho 1 file XML có cấu trúc và dữ liệu như dưới đây (employees.xml):

```
1 <?xml version="1.0" encoding="UTF-8" ?>
2 <employees>
3   <employee>
4     <id>1</id>
5     <name>Trần Thanh Thảo</name>
6   </employee>
7   <employee>
8     <id>2</id>
9     <name>Trương Hoài Phan</name>
10  </employee>
11  <employee>
12    <id>3</id>
13    <name>Nguyễn Quang Phúc</name>
14  </employee>
15  <employee>
16    <id>4</id>
17    <name>Phạm Thị Xuân Diệu</name>
18  </employee>
19  <employee>
20    <id>5</id>
21    <name>Hồ Trung Thành</name>
22  </employee>
23 </employees>
```

Bằng các kiến thức được cung cấp trong bài học, Sinh viên hãy viết phần mềm quản lý Employee, với cấu trúc dữ liệu như file XML ở trên.

Chương trình cần có các chức năng:

- (1) Mô hình hóa hướng đối tượng
- (2) Chức năng CRUD: Xem, thêm, sửa, xóa dữ liệu Employee
- (3) Thủ nghiệm các chức năng trong Console
- (4) Thiết kế giao diện tương tác người dùng với PyQt6 – Qt Designer:



Hướng dẫn:

Sinh viên xem lại kiến thức được trình bày trong môn học và tìm hiểu thêm trên mạng để thực hiện bài làm.

Sinh viên thực hiện lại các mã lệnh mẫu dưới đây để có thể: Xem, thêm, sửa, xóa dữ liệu trong XML. Từ đó vận dụng để làm các câu còn lại cho bài tập.

Bước 1: Tạo lớp hướng đối tượng Employee trong file “Employee.py”

```
1 class Employee:  
2     def __init__(self,id,name):  
3         self.id=id  
4         self.name=name  
5     def __str__(self):  
6         return f"{self.id}\t{self.name}"
```

Bước 2: Tạo file đọc toàn bộ dữ liệu trong file XML “TestReadXML.py”

```
1 import xml.etree.ElementTree as ET  
2 from Ex81.Employee import Employee  
3 tree = ET.parse("employees.xml")  
4 root = tree.getroot()  
5 for item in root.findall('employee'):  
6     id = item.find('id').text  
7     name = item.find('name').text  
8     emp=Employee(int(id),name)  
9     print(emp)
```

Ta chạy mã lệnh ở bước 2 này sẽ có kết quả như sau:

```
C:\Python\Python312\python.exe
1 Trần Thanh Thần
2 Trương Hoài Phan
3 Nguyễn Quang Phúc
4 Phạm Thị Xuân Diệu
5 Hồ Trung Thành
```

Bước 3: Tạo file thêm mới dữ liệu cho file XML “TestInsertXML.py”

```
1 import xml.etree.ElementTree as ET
2 tree = ET.parse("employees.xml")
3 root = tree.getroot()
4
5 emp = ET.Element("employee")
6
7 empId = ET.Element("id")
8 empId.text='113'
9 emp.append(empId)
10
11 empName = ET.Element("name")
12 empName.text="Trần Duy Thành"
13 emp.append(empName)
14
15 root.append(emp)
16
17 tree.write('employees.xml', encoding='utf-8')
```

Chạy mã lệnh trên để tạo một Element mới cho XML, sau đó quay lại bước 2 để xem lại kết quả. Ta có kết quả:

```
C:\Python\Python312\python.exe E
1 Trần Thanh Thần
2 Trương Hoài Phan
3 Nguyễn Quang Phúc
4 Phạm Thị Xuân Diệu
5 Hồ Trung Thành
113 Trần Duy Thành
```

Bước 4: Tạo file chỉnh sửa dữ liệu cho file XML “TextEditXML.py”

```
1 import xml.etree.ElementTree as ET
2 tree = ET.parse("employees.xml")
3 root = tree.getroot()
4 idSearch=4
5 newName="Trần Phạm Mẫn Nhi"
6 for item in root.findall('employee'):
7     id = item.find('id').text
8     if int(id) ==idSearch:
9         name = item.find('name')
10        name.text=newName
11 tree.write('employees.xml', encoding='utf-8')
```

Mã lệnh trên sẽ chỉnh sửa Employee có id =4. Ta chạy mã lệnh trên sau đó quay lại chạy TestReadXML.py để xem kết quả:

```
C:\Python\Python312\python.exe
1 Trần Thanh Thần
2 Trương Hoài Phan
3 Nguyễn Quang Phúc
4 Trần Phạm Mẫn Nhi
5 Hồ Trung Thành
113 Trần Duy Thanh
```

Bước 5: Tạo file xóa dữ liệu trong file XML “TestRemoveXML.py”

```
1 import xml.etree.ElementTree as ET
2 tree = ET.parse("employees.xml")
3 root = tree.getroot()
4 idSearch=4
5 itemDelete=None
6 for item in root.findall('employee'):
7     id = int(item.find('id').text)
8     if id ==idSearch:
9         itemDelete=item
10        break
11 if itemDelete!=None:
12     root.remove(itemDelete)
13     tree.write('employees.xml', encoding='utf-8')
```

Sinh viên thử nghiệm chức năng xóa, sau đó kiểm tra lại kết quả.

Câu 82: Chương trình quản lý Product – XML

Yêu cầu:

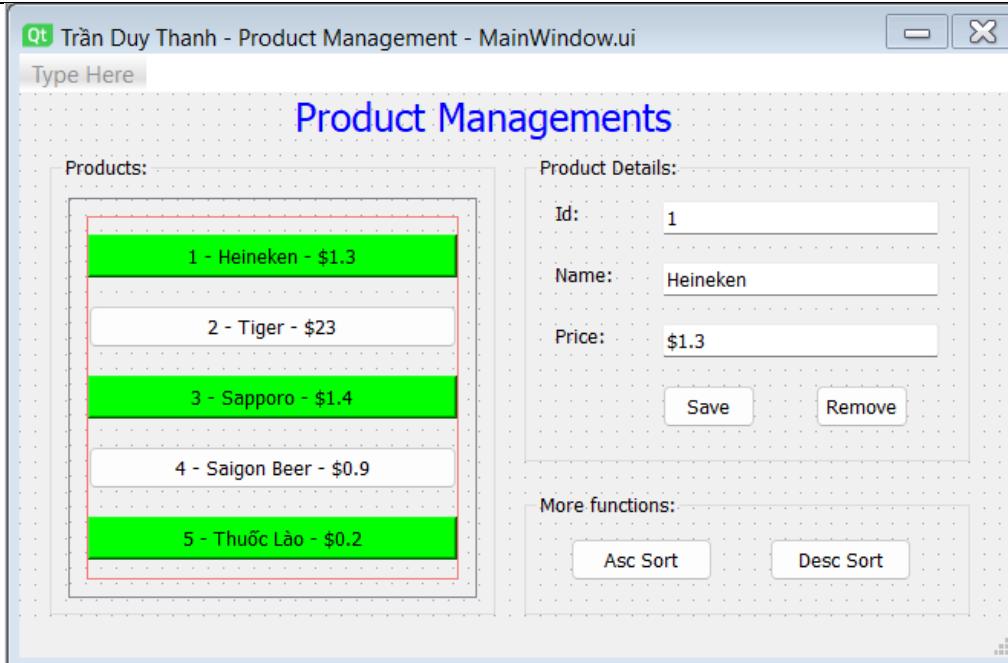
Cho 1 file XML có cấu trúc và dữ liệu như dưới đây (products.xml):

```
1 <products>
2   <product>
3     <id>1</id>
4     <name>Heineken</name>
5     <price>$1.5</price>
6   </product>
7   <product>
8     <id>2</id>
9     <name>Tiger</name>
10    <price>$1.3</price>
11  </product>
12  <product>
13    <id>3</id>
14    <name>Sapporo</name>
15    <price>$1.4</price>
16  </product>
17 </products>
```

Bằng các kiến thức được cung cấp trong bài học, và tham khảo ở bài tập trước, Sinh viên hãy viết phần mềm quản lý Product, với cấu trúc dữ liệu như file XML ở trên.

Chương trình cần có các chức năng:

- (1) Mô hình hóa hướng đối tượng
- (2) Chức năng CRUD: Xem, thêm, sửa, xóa dữ liệu Product
- (3) Thủ nghiệm các chức năng trong Console
- (4) Thiết kế giao diện tương tác người dùng với PyQt6 – Qt Designer:
(5*) Đưa thẻ `<id>` thành thuộc tính của Product và coding lại phần tương tác XML. Tức là `<product id=1>...</product>`. Nên áp dụng kiến trúc kế thừa để không làm ảnh hưởng tới các coding cũ đã chạy trước đó.



Hướng dẫn:

Sinh viên xem lại kiến thức được trình bày trong môn học và tìm hiểu thêm trên mạng để thực hiện bài làm. Tham khảo bài Employee - XML để xử lý.

Câu 83: Chuyển đổi String JSON qua Python Object

Yêu cầu:

Cho string Json có cấu trúc sau:

```
{ "Id":"Emp1", "Age":50, "Name":"Trần Duy Thành"}
```

Hãy viết mã lệnh chuyển đổi qua Python Object

Hướng dẫn:

Sinh viên xem lại kiến thức được trình bày trong môn học và tìm hiểu thêm trên mạng để thực hiện bài làm. Xem mã lệnh mẫu:

```
1 import json
2 jsonString = '{ "Id":"Emp1", "Age":50, "Name":"Trần Duy Thành"}'
3 dataObject=json.loads(jsonString)
4 print(dataObject)
5 print("ID=",dataObject["Id"])
6 print("Age=",dataObject["Age"])
7 print("Name=",dataObject["Name"])
```

Câu 84: Chuyển đổi Python Object qua String JSON

Yêu cầu:

Cho Python Object có cấu trúc sau:

```
pythonObject = {  
    "ten": "Trần Duy Thanh",  
    "tuoi": 50,  
    "ma": "nv1"  
}
```

Hãy viết mã lệnh chuyển đổi qua String json.

Hướng dẫn:

Sinh viên xem lại kiến thức được trình bày trong môn học và tìm hiểu thêm trên mạng để thực hiện bài làm. Mã lệnh mẫu:

```
1 import json  
2 pythonObject = {  
3     "Name": "Trần Duy Thanh",  
4     "Age": 50,  
5     "Id": "Emp1"  
6 }  
7 jsonString = json.dumps(pythonObject, ensure_ascii=False)  
8 # the result is a JSON string:  
9 print(jsonString)
```

Câu 85: Chương trình quản lý Assets – JSON (*)

Yêu cầu:

Viết một phần mềm quản lý tài sản (Assets) có mô tả như sau:

Employee: EmployeeId, EmployeeName, UserName, Password

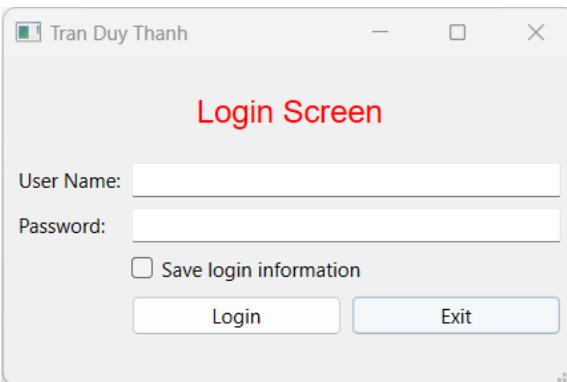
Asset: AssetId, AssetName, ImportYear, Value

Employee_Asset: EmployeeId, AssetId, Role

Phần mềm cần có 2 màn hình: Màn hình đăng nhập và màn hình quản trị tài sản.

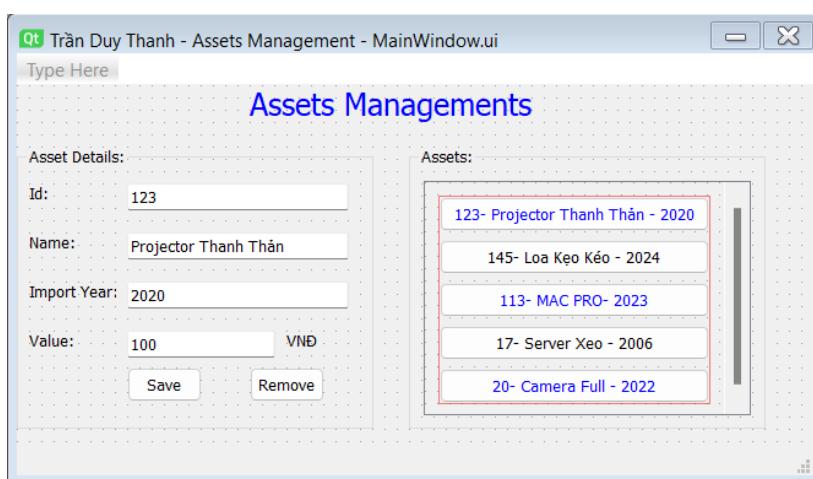
Sinh viên cần thực hiện các yêu cầu sau:

- (a) Tự đề xuất cấu trúc JSON cho 3 file dữ liệu Employee, Asset và Employee_Asset. Tiến hành nhập dữ liệu mẫu cho 3 file này, mỗi file tối thiểu 5 dữ liệu liên quan.
- (b) Thiết kế các class đối tượng cho Employee, Asset và Employee_Asset. Khai báo các thuộc tính và kiểu dữ liệu phù hợp theo yêu cầu.
- (c) Sử dụng các mã lệnh phù hợp để tương tác dữ liệu trong JSON và mapping các đối tượng.
- (d) Khi chạy chương trình, mặc định màn hình đăng nhập sẽ xuất hiện, đăng nhập thành công thì mở ra màn hình quản trị tài sản:



Màn hình đăng nhập mẫu

Bấm Login để đăng nhập, đăng nhập thành công sẽ hiển thị quản trị tài sản:



Hướng dẫn:

Sinh viên xem lại kiến thức được trình bày trong môn học và tìm hiểu thêm trên mạng để thực hiện bài làm.

Câu 86: Xử lý dữ liệu trong tập tin CSV

Yêu cầu:

Cho cấu trúc dữ liệu CSV employee.csv như sau:

1	Id,Name,Dob,Role
2	1,Tuấn Kiệt,12/02/2000,Web Developer
3	2,Khánh Hưng,22/04/2003,Tester
4	3,Gia Hân,06/08/2002,Business Analyst
5	4,Ngọc Tú,02/02/2001,Mobile App Developer
6	

<https://tranduythanh.com/datasets/employee.csv>

Hãy thực hiện các chức năng đọc và ghi tập tin CSV ở trên (lưu ý dòng cuối cùng là một chuỗi rỗng).

Hướng dẫn:

Sinh viên xem lại kiến thức được trình bày trong môn học và tìm hiểu thêm trên mạng để thực hiện bài làm.

Tạo file “TestReadCSV.py” để test đọc CSV (lưu ý Sinh viên có thể dùng Pandas):

```
1 import csv
2 with open('employee.csv', newline='', encoding="utf8") as f:
3     reader = csv.reader(f, delimiter=',', quoting=csv.QUOTE_NONE)
4     for row in reader:
5         print(' - '.join(row))
6         # print(row[0] + ' * ' + row[1] + ' * ' + row[2] + ' * ' +
7 row[3])
```

Kết quả khi chạy mã lệnh đọc CSV:

The screenshot shows a Jupyter Notebook interface with a single cell containing Python code. The code reads a CSV file named 'employee.csv' and prints each row as a string where the elements are separated by a dash ('-'). The output of the cell is the same data structure as the table provided in the question, showing four rows of employee information.

```
Run TestReadCSV
```

```
C:\Python\Python312\python.exe E:\Elearning\Prog
Id - Name - Dob - Role
1 - Tuấn Kiệt - 12/02/2000 - Web Developer
2 - Khánh Hưng - 22/04/2003 - Tester
3 - Gia Hân - 06/08/2002 - Business Analyst
4 - Ngọc Tú - 02/02/2001 - Mobile App Developer
```

Tạo file “TestWriteCSV.py” để ghi dữ liệu vào CSV:

```
1 import csv
2
3 with open('employee.csv', mode='a+',
4             encoding="utf8", newline='') as f:
5     employee_writer = csv.writer(f, delimiter=',', quotechar='"')
6     employee_writer.writerow(['5', 'Tú Linh', '02/02/2002'])
7     employee_writer.writerow(['6', 'Nam Giao', '03/04/2000'])
8     employee_writer.writerow(['7', 'Huỳnh Anh', '05/11/2001'])
9
```

Chạy mã lệnh trên ta sẽ có các dữ liệu được lưu nối đuôi vào file CSV gốc.

Câu 87: Chương trình book Hotel – CSV (*)

Yêu cầu:

Viết minh họa một phần chức năng đặt phòng trong phần mềm Book Hotel cho công ty *Thanh Thản Tourist* như mô tả dưới đây:

Khách hàng: mã khách hàng, tên khách hàng, phone, email

Phòng: mã phòng, tên phòng, loại phòng (VIP/Thường).

Đặt phòng: mã khách hàng, mã phòng, ngày bắt đầu ở, ngày kết thúc ở.

Sinh viên cần thực hiện các yêu cầu sau:

(a) Tạo các file dữ liệu CSV tương ứng với các đối tượng được mô tả ở trên. Mỗi file dữ liệu nhập từ 5 tới 10 đối tượng để thử nghiệm.

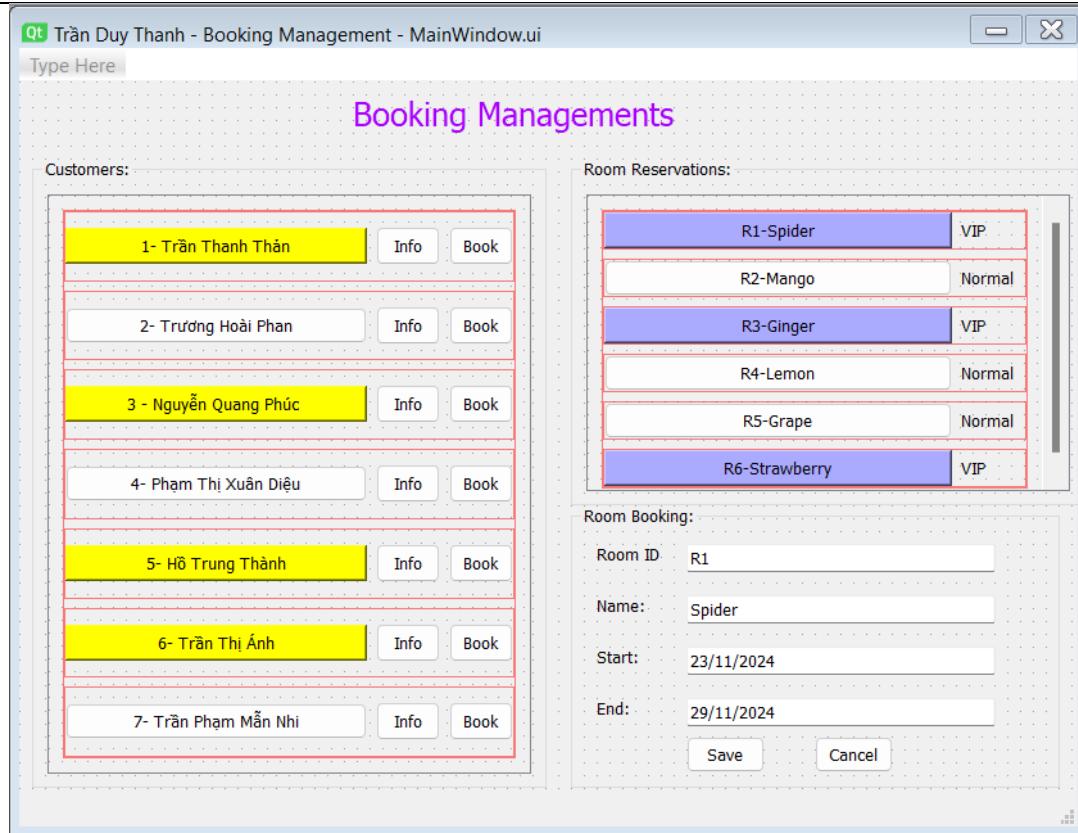
(b) Mô hình hóa hướng đối tượng, tạo các lớp phù hợp với yêu cầu

(c) Viết các hàm chức năng để đáp ứng CRUD: Xem, thêm, sửa, xóa các đối tượng liên quan trọng hệ thống.

(d) Viết các lớp xử lý lưu file và đọc file riêng biệt cho từng đối tượng Khách hàng, phòng, đặt phòng. Có thể nghĩ tới kỹ thuật kế thừa để tái sử dụng các mã lệnh của phần mềm.

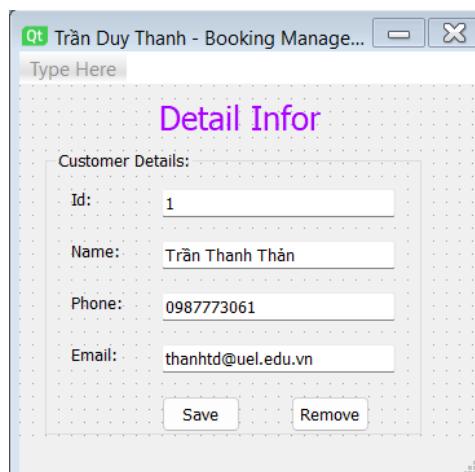
(e) Thủ nghiệm các chức năng trong Console

(f*) Thiết kế giao diện tương tác người dùng để thực nghiệm các chức năng của hệ thống:



Ở trên là màn hình chi tiết của một phần chức năng đặt phòng Hotel. Mô tả sơ lược các chức năng như sau:

- (f1*) Khi khởi động phần mềm, toàn bộ Khách hàng trong CSV sẽ được nạp vào danh sách “Customers”, cách nạp xem lại các bài trước. Mỗi dòng giao diện chính là 1 QHBoxLayout(có 3 QPushButton: Thông tin khách hàng, Info và Book)
- (f2*) Khi khởi động phần mềm, toàn bộ Phòng trong CSV sẽ được nạp vào danh sách “Room Reservations”, cách làm như f1*
- (f3*) Khi nhấp vào nút “Infor” của customer, thì thông tin chi tiết sẽ được hiển thị vào giao diện dưới đây: Cung cấp chức năng cập nhật dữ liệu và xóa khách hàng.



(f4*) Khi nhấn vào nút “Book” của customer, nếu Customer đã đăng ký phòng trước đó, thì thông tin chi tiết của phòng được đăng ký sẽ hiển thị vào mục “Room Booking”.

(f5*) Người sử dụng có thể chọn “Cancel” để hủy đặt phòng nếu trước đó đã đặt

(f6*) Nếu muốn đặt phòng khác, thì người sử dụng chọn Phòng từ danh sách “Room Reservations”, khi nhấn vào từng Phòng thì thông tin chi tiết của nó sẽ được hiển thị xuống mục “Room Booking”, người sử dụng xem thông tin phòng và nhập các dữ liệu đặt phòng rồi nhấn “Save” để thực hiện thao tác đặt phòng.

Hướng dẫn:

Sinh viên xem lại kiến thức được trình bày trong môn học và tìm hiểu thêm trên mạng để thực hiện bài làm.

Bắt buộc phải thực hiện hoàn chỉnh câu (a) tới câu (e). Câu (f) là câu nâng cao.

Câu 88: Xử lý dữ liệu trong tập tin Excel XLSX

Yêu cầu:

Hãy dùng thư viện xlsxwriter để lưu ra file Excel có cấu trúc như dưới đây(bao gồm cả logo):

STT	MÃ SẢN PHẨM	TÊN SẢN PHẨM	SỐ LƯỢNG	ĐƠN GIÁ
1	SP1	Coca	15	15000
2	SP2	Pepsi	20	18000

Hướng dẫn:

Sinh viên xem lại kiến thức được trình bày trong môn học và tìm hiểu thêm trên mạng để thực hiện bài làm.

Bước 1: Viết các mã lệnh để lưu trữ dữ liệu vào tập tin Excel.

```
1 import xlsxwriter
2 # Tạo một file excel cùng 1 sheet
3 workbook = xlsxwriter.Workbook('demo.xlsx')
4 worksheet = workbook.add_worksheet()
5 # thiết lập các cột cho file
6 worksheet.set_column('A:A', 5)
7 worksheet.set_column('B:B', 15)
8 worksheet.set_column('C:C', 20)
9 worksheet.set_column('D:D', 15)
10 worksheet.set_column('E:E', 15)
11 # định dạng tiêu đề cột in đậm
12 bold = workbook.add_format({'bold': True})
13
14 # thêm dòng tiêu đề và định dạng in đậm
15 worksheet.write('A1', 'STT', bold)
16 worksheet.write('B1', 'MÃ SẢN PHẨM', bold)
17 worksheet.write('C1', 'TÊN SẢN PHẨM', bold)
18 worksheet.write('D1', 'SỐ LƯỢNG', bold)
19 worksheet.write('E1', 'ĐƠN GIÁ', bold)
20 #thêm một dòng dữ liệu
21 worksheet.write('A2', 1)
22 worksheet.write('B2', 'SP1')
23 worksheet.write('C2', 'Coca')
24 worksheet.write('D2', '15')
25 worksheet.write('E2', '15000')
26 #thêm một dòng dữ liệu
27 worksheet.write('A3', 2)
28 worksheet.write('B3', 'SP2')
29 worksheet.write('C3', 'Pepsi')
30 worksheet.write('D3', '20')
31 worksheet.write('E3', '18000')
32 #Chèn Logo vào
33 worksheet.insert_image('B5', 'logo_UEL.png')
34 workbook.close()
```

Lưu ý hình Logo Sinh viên chép cùng nơi lưu source code.

Sau khi chạy mã lệnh trên thì ta có kết quả như mong muốn.

Bước 2: Viết mã lệnh để đọc file Excel ở bước 1

```
1 from openpyxl import load_workbook
2 wb = load_workbook('demo.xlsx')
3 print(wb.sheetnames)
4 ws = wb[wb.sheetnames[0]]
5 for row in ws.values:
6     for value in row:
7         print(value, "\t", end="")
8     print("")
```

Câu 89: Viết phần mềm quản lý Nhân viên - Excel**Yêu cầu:**

Viết phần mềm quản lý Nhân viên lưu bằng Excel. Mỗi nhân viên có Mã, Tên, Tuổi.

- (a) Phần mềm cho phép lưu Nhân viên vào File Excel
- (b) Phần mềm cho phép đọc danh sách Nhân viên trong File Excel
- (c) Phần mềm cho phép sắp xếp Nhân viên theo Tuổi tăng dần

Cấu trúc của File Excel như sau:

	A	B	C	D
1	STT	Mã	Tên	Tuổi
2	1	NV1	An	18
3	2	NV2	Lành	22
4	3	NV3	Giải	20
5	4	NV4	Thoát	19
6	5	NV5	Hạnh	25
7	6	NV6	Phúc	24
8				

Yêu cầu phải mô hình hóa đối tượng, thử nghiệm các hàm trong console sau đó tiến hành để xuất thiết kế giao diện tương tác.

Hướng dẫn:

Sinh viên xem lại kiến thức được trình bày trong môn học và tìm hiểu thêm trên mạng để thực hiện bài làm.

Câu 90: Phần mềm quản lý thiết bị XML (*)

Yêu cầu:

Chương trình quản lý thiết bị gồm có 2 tập dữ liệu như mô tả dưới đây.

Tập lưu danh sách nhóm thiết bị có tên **nhomthietbi.xml** có dữ liệu mẫu và format như sau:

```
1 <?xml version="1.0" encoding="UTF-8" ?>
2 <nhoms>
3   <nhom>
4     <ma>n1</ma>
5     <ten>Nhóm 1</ten>
6   </nhom>
7   <nhom>
8     <ma>n2</ma>
9     <ten>Nhóm 2</ten>
10  </nhom>
11  <nhom>
12    <ma>n3</ma>
13    <ten>Nhóm 3</ten>
14  </nhom>
15 </nhoms>
```

Theo cấu trúc ở trên thì mỗi Nhóm sẽ có: Mã nhóm, tên nhóm.

Chương trình phải đọc dữ liệu danh sách nhóm thiết bị.

Tập dữ liệu thiết bị được lưu trong file **ThietBi.xml**, có dữ liệu và cấu trúc như sau:

```
1 <?xml version="1.0" encoding="UTF-8" ?>
2 <thietbis>
3   <thietbi manhom="n1">
4     <ma>tb1</ma>
5     <ten>Thiết bị 2</ten>
6   </thietbi>
7   <thietbi manhom="n1">
8     <ma>tb2</ma>
9     <ten>Thiết bị 2</ten>
10  </thietbi>
11  <thietbi manhom="n2">
```

```
12      <ma>tb3</ma>
13      <ten>Thiết bị 3</ten>
14  </thietbi>
15  <thietbi manhom="n3">
16      <ma>tb4</ma>
17      <ten>Thiết bị 4</ten>
18  </thietbi>
19  <thietbi manhom="n3">
20      <ma>tb5</ma>
21      <ten>Thiết bị 5</ten>
22  </thietbi>
23 </thietbis>
```

Theo cấu trúc ở trên, thì mỗi Thiết bị sẽ có thuộc tính Mã nhom để phân loại thiết bị vào đúng nhóm thiết bị. Ngoài ra mỗi thiết bị sẽ có thêm 2 thuộc tính: mã và Tên.

Chương trình cần cung cấp các chức năng:

- (a) Hiển thị danh sách Nhóm thiết bị
- (b) Hiển thị toàn bộ Thiết bị
- (c) Lọc Danh sách Thiết bị theo Nhóm thiết bị
- (d) Xuất Nhóm thiết bị có số lượng thiết bị nhiều nhất

Hướng dẫn:

Sinh viên xem lại kiến thức được trình bày trong môn học và tìm hiểu thêm trên mạng để thực hiện bài làm.

Câu 91: Quản lý Sản phẩm - Pickle File

Yêu cầu:

Viết phần mềm Quản Lý sản phẩm

Mỗi danh mục có: Mã , tên; Một danh mục có nhiều sản phẩm

Mỗi sản phẩm có: Mã, tên, đơn giá; Mỗi một sản phẩm thuộc về một danh mục.

Hãy thiết kế các lớp mô hình: Lớp Danh Mục, Lớp Sản Phẩm, Lớp FileFactory.

Khai báo các thuộc tính và phương thức phù hợp, vì Danh Mục có nhiều sản phẩm nên cần sử dụng collection để lưu trữ.

Sau đó thực hiện các chức năng:

- (a) Các chức năng CRUD đối với Danh mục
- (b) Các chức năng CRUD đối với Sản phẩm
- (c) Mọi thay đổi dữ liệu đều được lưu xuống định dạng Pickle File thông qua lớp FileFactory.
- (d) Viết các mã lệnh để thực nghiệm các chức năng của phần mềm.

Hướng dẫn:

Sinh viên xem lại kiến thức được trình bày trong môn học và tìm hiểu thêm trên mạng để thực hiện bài làm. Dưới đây là coding minh họa, Sinh viên tham khảo để viết các chức năng còn lại của chương trình:

Bước 1: Tạo lớp Product trong file Product.py

```
1 class Product:  
2     def __init__(self,id=None,name=None,price=None):  
3         self.id=id  
4         self.name=name  
5         self.price=price  
6     def __str__(self):  
7         return f"{self.id}\t{self.name}\t{self.price}"
```

Bước 2: Tạo lớp Category trong file Category.py

```
1 class Category:  
2     def __init__(self,id=None,name=None,products=None):  
3         if products==None:  
4             self.products=[]  
5         else:  
6             self.products=products  
7         self.id=id  
8         self.name=name  
9     def __str__(self):  
10        return f"{self.id}\t{self.name}"  
11    def addProduct(self,product):  
12        self.products.append(product)  
13    def printProducts(self):  
14        for product in self.products:  
15            print(product)
```

Bước 3: Tạo lớp FileUtils trong FileUtils.py để Serialize và Deserialize đối tượng:

```
1 import pickle
2 class FileUtils:
3     @staticmethod
4     def savemodel(model,filename):
5         try:
6             pickle.dump(model, open(filename, 'wb'))
7             return True
8         except:
9             print("An exception occurred")
10            return False
11    @staticmethod
12    def loadmodel(filename):
13        try:
14            model=pickle.load(open(filename, 'rb'))
15            return model
16        except:
17            print("An exception occurred")
18            return None
```

Bước 4: Tạo TestWritePickleDataApp.py để thử nghiệm chức năng tạo các đối tượng Danh mục và sản phẩm, cùng các mối quan hệ của chúng, Đồng thời Serialize đối tượng xuống ổ cứng và xuất dữ liệu ra cho từng đối tượng và toàn bộ đối tượng để kiểm chứng:

```
1 from Ex90.Category import Category
2 from Ex90.FileUtil import FileUtils
3 from Ex90.Product import Product
4
5 dataset=[]
6
7 cate1=Category("cate1","Electronic")
8 cate1.addProduct(Product("p1","Light X",100))
9 cate1.addProduct(Product("p2","Fan X",200))
10 cate1.addProduct(Product("p3","MIC Z",150))
11 print("Cate 1 information:")
12 print(cate1)
13 print("-"*15)
14 cate1.printProducts()
15 print("-"*15)
16 dataset.append(cate1)
17
```

```
18 cate2=Category("cate2","Houseware")
19 cate2.addProduct(Product("p4","Bowl",20))
20 cate2.addProduct(Product("p5","Knife",10))
21 cate2.addProduct(Product("p6","Pan",80))
22 print("Cate 2 information:")
23 print(cate2)
24 print("-"*15)
25 cate2.printProducts()
26 print("-"*15)
27
28 dataset.append(cate2)
29 print("All Categories:")
30 for cate in dataset:
31     print(cate)
32     cate.printProducts()
33
34 ret=FileUtil.savemodel(dataset,"mydataset.data")
35 if ret==True:
36     print("Save Pickle File successful!")
```

Chạy các mã lệnh trong bước 4 ta có kết quả:

```
Run  TestWritePickleDataApp (1)  Open Selected Qt Des
C:\Python\Python312\python.exe E:\E1
Cate 1 information:
cate1  Electronic
-----
p1  Light X 100
p2  Fan X  200
p3  MIC Z  150
-----
Cate 2 information:
cate2  Houseware
-----
p4  Bowl    20
p5  Knife   10
p6  Pan 80
-----
All Categories:
cate1  Electronic
p1  Light X 100
p2  Fan X  200
p3  MIC Z  150
cate2  Houseware
p4  Bowl    20
p5  Knife   10
p6  Pan 80
Save Pickle File successful!
```

Lúc này file Pickle cũng được tạo ra, nó là toàn bộ dữ liệu của đối tượng, được lưu trong file “**mydataset.data**”

Bước 5: Tạo **TestReadPickleDataApp.py** để thử nghiệm chức năng Deserialize đối tượng:

```
1 from Ex90.FileUtil import FileUtil
2
3 dataset=FileUtil.loadmodel("mydataset.data")
4 for cate in dataset:
5     print(cate)
6     cate.printProducts()
7
```

Chạy mã lệnh ở Bước 5, ta kết quả như dưới đây:

```
Run TestReadPickleDataApp x Open Selected Qt Designer x
cate1 Electronic
p1 Light X 100
p2 Fan X 200
p3 MIC Z 150
cate2 Houseware
p4 Bowl 20
p5 Knife 10
p6 Pan 80
```

Như vậy Sinh viên đã có hầu hết các mã lệnh quan trọng của bài tập Pickle này, dựa vào các mã lệnh minh họa ở trên để tiếp tục vận dụng xử lý các chức năng còn lại của chương trình.

Câu 92: Phần mềm quản lý Sinh viên – Pickle File(*)

Yêu cầu:

Viết phần mềm quản lý Sinh Viên

Mỗi một lớp học có: Mã lớp, tên; một lớp có nhiều Sinh viên

Mỗi sinh viên có: mã, tên, năm sinh; Mỗi một sinh viên thuộc về một lớp.

Môn học có: Mã môn học, tên môn học, số tín chỉ

Lớp học phần có: Mã lớp học phần, mã môn học, số tín chỉ, ngày bắt đầu, ngày kết thúc

Một Sinh viên có thể đăng ký nhiều lớp học phần, và một lớp học phần có thể được đăng ký bởi nhiều Sinh viên.

Yêu cầu thiết kế mô hình lớp hướng đối tượng cho phần mềm quản lý Sinh viên với mô tả ở trên.

Cung cấp các chức năng để: Lưu mới, sửa, xóa, tìm kiếm, sắp xếp, lưu và đọc dữ liệu của các đối tượng bằng Pickle File.

Chương trình chính cần có các menu:

- (a) Xem danh sách lớp học
- (b) Xem danh sách sinh viên của từng lớp học
- (c) Xem danh sách môn học
- (d) Xem danh sách lớp học phần mà sinh viên đã đăng ký học
- (e) Xuất danh sách Sinh viên theo từng lớp học phần đã đăng ký
- (f) Xóa Sinh viên khỏi lớp học phần
- (g) Lưu Pickle File
- (h) Đọc Pickle File

Hướng dẫn:

Sinh viên xem lại kiến thức được trình bày trong môn học và tìm hiểu thêm trên mạng để thực hiện bài làm.

Câu 93: Xử lý đa phương tiện

Yêu cầu:

Với các bài tập quản lý sản phẩm, Sinh viên bổ sung thêm thuộc tính hình ảnh và tiến hành xử lý hiển thị hình ảnh, thay đổi hình ảnh cho các sản phẩm.

Hướng dẫn:

Sinh viên xem lại kiến thức được trình bày trong môn học và tìm hiểu thêm trên mạng để thực hiện bài làm.

MODULE 5

THIẾT KẾ GIAO DIỆN TƯƠNG TÁC NGƯỜI DÙNG NÂNG CAO

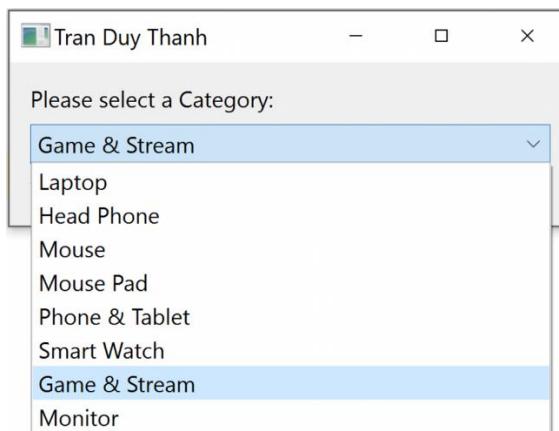
Nội dung kiến thức thực hành:

- + Ôn tập PyQt6 & Qt Designer: Cách tạo giao diện cơ bản, cách xử lý sự kiện cho các Widgets
- + Thực hành về xử lý giao diện dạng danh sách QComboBox
- + Thực hành về xử lý giao diện dạng danh sách QListWidget
- + Thực hành về các Widgets thời gian: QDateEdit, QTimeEdit, QDateTimeEdit
- + Thực hành về xử lý giao diện dạng bảng QTableWidget
- + Thực hành về các loại Dialogs: Các Dialog xác thực, các Dialog nhập liệu
- + Thực hành về OpenFileDialog và SaveFileDialog
- + Mapping hướng đối tượng lên giao diện, tương tác đối tượng giữa giao diện và tập tin trên ổ cứng để lưu trữ và hiển thị dữ liệu đối tượng.

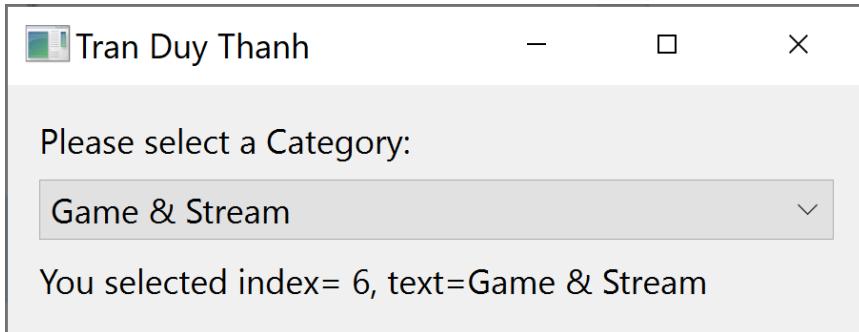
Câu 94: Nạp dữ liệu và tương tác QComboBox

Yêu cầu:

Thiết kế giao diện để nạp dữ liệu và tương tác QComboBox như hình dưới đây:



Gọi hàm addItem, insertItems của đối tượng QCombobox để thêm các dữ liệu vào giao diện này. Khi nhấn chọn các phần tử trong QCombobox thì chương trình sẽ thông báo người sử dụng đang chọn phần tử ở vị trí bao nhiêu, nhấn lựa chọn là gì:



Đối với xử lý sự kiện chọn Item trong QCombobox, Sinh viên tìm hiểu signal **activated** để xử lý.

Hướng dẫn:

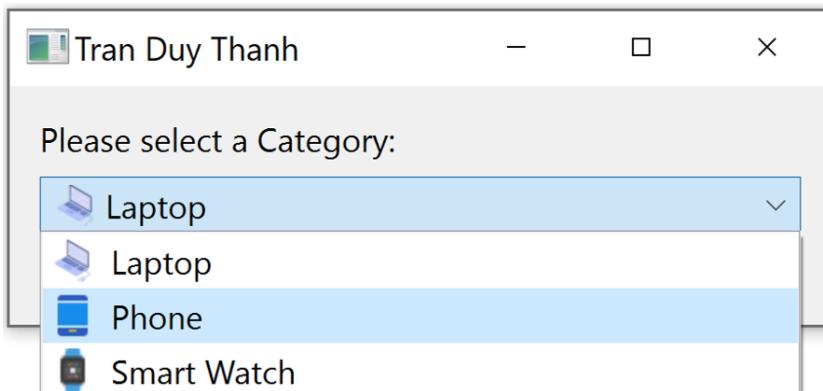
Sinh viên xem hướng dẫn chi tiết tại bài giảng:

<https://tranduythanh.com/2023/11/27/bai-16-qcombobox-basic-widgets-pyqt6/>

Câu 95: Nạp dữ liệu hướng đối tượng và tương tác QCombobox

Yêu cầu:

Sinh viên mở rộng bài tập QCombobox ở bài trước, sử dụng hướng đối tượng để xử lý Category mapping lên giao diện:



Sinh viên giả lập một số đối tượng Category cùng với hình ảnh của nó để hiển thị lên giao diện.

Hướng dẫn:

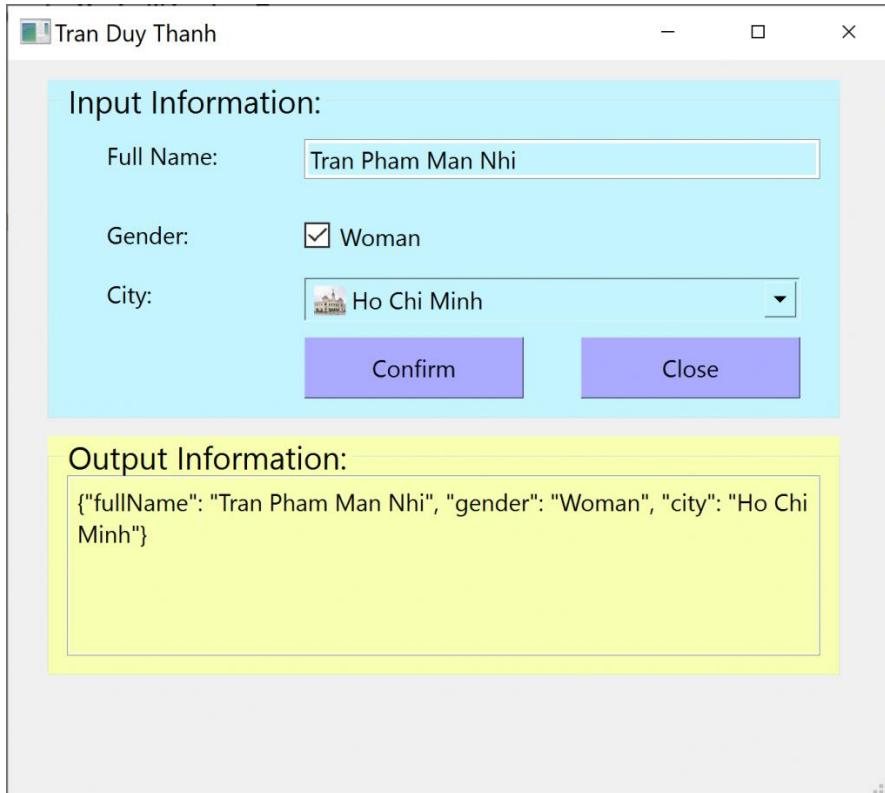
Sinh viên xem hướng dẫn chi tiết tại bài giảng:

<https://tranduythanh.com/2023/11/27/bai-16-qcombobox-basic-widgets-pyqt6/#customqcombobox>

Câu 96: Quản lý thông tin Nhân viên

Yêu cầu:

Kết hợp các Widgets cơ bản và QCombobox để thiết kế phần mềm quản lý thông tin Nhân viên như dưới đây:



Sinh viên cần thiết kế các class hướng đối tượng để mapping lên giao diện.

Hướng dẫn:

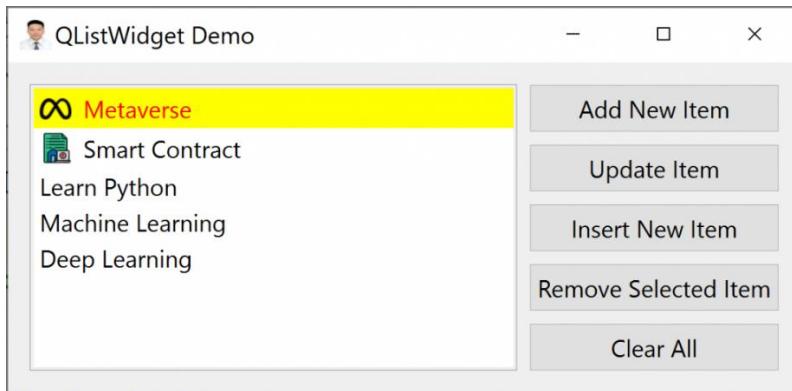
Sinh viên xem hướng dẫn chi tiết tại bài giảng:

<https://tranduythanh.com/2023/11/27/bai-16-qcombobox-basic-widgets-pyqt6/#advancedcombobox>

Câu 97: Nạp dữ liệu và tương tác QListWidget

Yêu cầu:

Thiết kế giao diện để nạp dữ liệu và tương tác QListWidget như hình dưới đây:



Chương trình gồm 8 chức năng chính:

- (1) Chức năng hiển thị danh sách mặc định ban đầu lên QListWidget, bao gồm các minh họa về Icon, màu nền, màu chữ...
- (2) Chức năng thêm mới một QListWidgetItem
- (3) Chức năng Cập nhật một Item
- (4) Chức năng Chèn một Item vào một vị trí bất kỳ trong QListWidget
- (5) Chức năng Xóa một Item ra khỏi QListWidget
- (6) Chức năng xóa toàn bộ Item trong QListWidget
- (7) Chức năng lựa chọn một Item bất kỳ trên QListWidget thì dữ liệu sẽ hiển thị lên Title của cửa sổ
- (8) Chức năng khi Double click vào một Item bất kỳ trong QListWidget thì sẽ hiển thị màn hình cập nhật dữ liệu cho Item này.

Hướng dẫn:

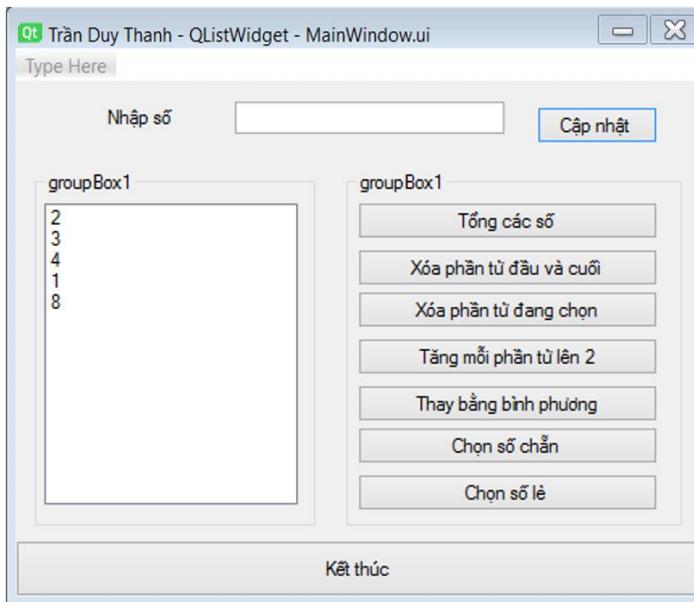
Sinh viên xem hướng dẫn chi tiết tại bài giảng:

<https://tranduythanh.com/2023/12/01/bai-17-qlistwidget-part-1-basic-widgets-pyqt6/>

Câu 98: Nạp dữ liệu và tương tác QListWidget

Yêu cầu:

Thiết kế giao diện để nạp dữ liệu và tương tác QListWidget như hình dưới đây:



Chương trình gồm 8 chức năng chính:

- (1) “Cập nhật”: Khi nhấn cập nhật thì số trong ô nhập sẽ đưa vào groupBox1
- (2) “Tổng các số”: Dùng QMessageBox để hiển thị tổng giá trị trong groupBox1
- (3) “Xóa phần tử đầu và cuối”: Các số đầu và cuối groupBox1 sẽ bị xóa
- (4) “Xóa phần tử đang chọn”: Số nào đang chọn trong groupBox1 sẽ bị xóa
- (5) “Tăng mỗi phần tử lên 2”: Toàn bộ số trong groupBox1 sẽ tăng lên 2
- (6) “Thay bằng bình phương”: Toàn bộ số trong groupBox1 sẽ tăng bình phương.
- (7) “Chọn số chẵn”: Các số nào là chẵn trong groupBox1 sẽ được tô vàng
- (8) “Chọn số lẻ”: Các số nào là lẻ trong groupBox1 sẽ được tô hồng. Lưu ý với chức năng tô màu thì phải bỏ chọn các số đã tô màu trước đó.
- (9) Nút “Kết thúc”: Sử dụng QMessageBox để xác thực có muốn thoát hay không.

Hướng dẫn:

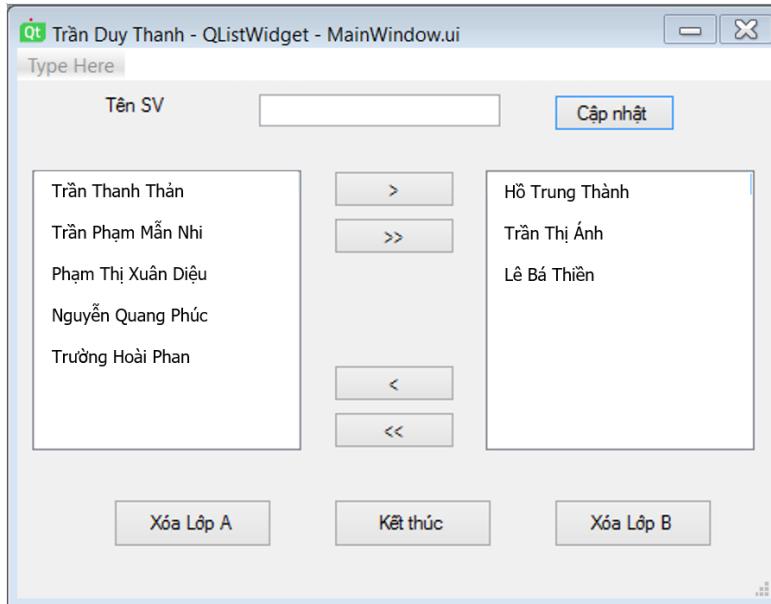
Sinh viên xem lại kiến thức được trình bày trong môn học và tìm hiểu thêm trên mạng để thực hiện bài làm. Coding mẫu minh họa xác thực thoát phần mềm:

```
1 import sys
2 from PyQt6.QtWidgets import QMessageBox
3 def processExit(self):
4     dlg = QMessageBox(self.MainWindow)
5     dlg.setWindowTitle("Confirmation Exit")
6     dlg.setText("Are you sure you want to Exit?")
7     buttons = QMessageBox.StandardButton.Yes |
8         QMessageBox.StandardButton.No
9     dlg.setStandardButtons(buttons)
10    button = dlg.exec()
11    if button == QMessageBox.StandardButton.Yes:
12        sys.exit(0)
```

Câu 99: Nạp dữ liệu và tương tác QListWidget

Yêu cầu:

Thiết kế giao diện để nạp dữ liệu và tương tác QListWidget như hình dưới đây:



Chương trình gồm các chức năng chính:

- (1) Khi người sử dụng Click “Cập nhật” hoặc nhấn phím Enter thì mặt định nhập tên sinh viên từ QLineEdit vào danh sách lớp A (không chấp nhận dữ liệu rỗng).

- (2) Các nút “>” và “<” khi được Click sẽ di chuyển tất cả các Sinh viên đang chọn sang QListWidget bên kia tương ứng.
- (3) Các nút “>>” và “<<” khi được Click sẽ di chuyển toàn các Sinh viên qua QListWidget bên kia tương ứng.
- (4) Nút lệnh “Xóa lớp A”, “Xóa lớp B” cho phép xóa các mục đang chọn trong QListWidget tương ứng.
- (5) Nút “Kết thúc”: Sử dụng QMessageBox để xác thực có muốn thoát hay không

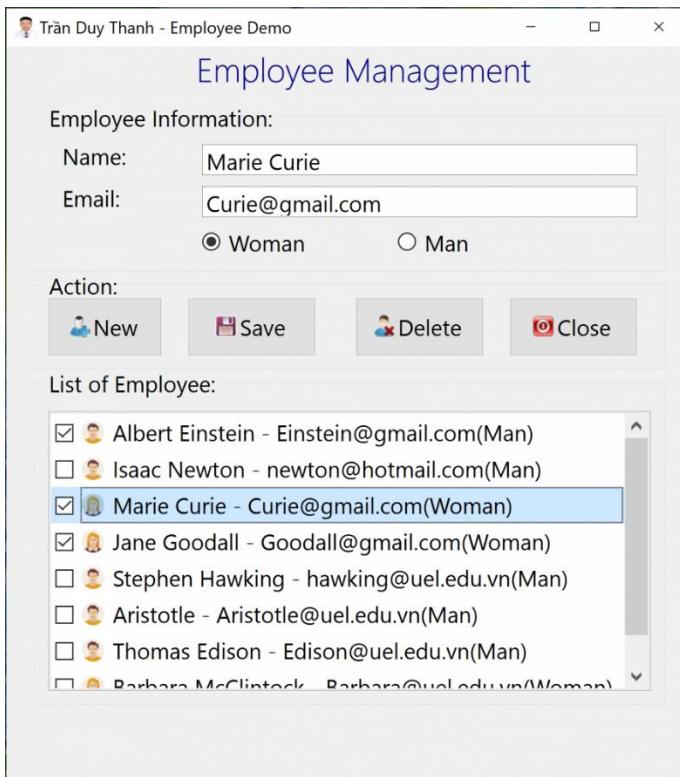
Hướng dẫn:

Sinh viên xem lại kiến thức được trình bày trong môn học và tìm hiểu thêm trên mạng để thực hiện bài làm.

Câu 100: Quản lý nhân viên - QListWidget

Yêu cầu:

Phối với các Widgets cơ bản và QListWidget cùng với xử lý JSON File để thiết kế phần mềm quản lý Nhân viên như hình dưới đây:



Mô tả sơ lược chức năng và nhiệm vụ Sinh viên phải thực hiện:

- (1) Chức năng mô hình hóa hướng đối tượng: Lớp Employee được thiết kế và được mô hình hóa dữ liệu từ giao diện vào danh sách hướng đối tượng Employee cũng như mapping thành định dạng JSON Array để lưu vào ổ cứng (gọi là Json Encode).
- (2) Chức năng phục hồi dữ liệu và mô hình hóa lại hướng đối tượng từ JSON Array được lưu trong ổ cứng lên bộ nhớ và được mapping và danh sách đối tượng Employee (gọi là Json Decode). Chức năng này sẽ hỗ trợ thao tác hiển thị lại dữ liệu danh sách Employee lên giao diện
- (3) Chức năng “New”: Khi nhấn vào button này thì các QLineEdit sẽ bỏ trống và con trỏ văn bản được focus tới ô “Name” để người sử dụng dễ dàng nhập liệu
- (4) Chức năng “Save”: Chức năng này có 2 nhiệm vụ là lưu mới Employee hoặc cập nhật Employee. Nếu Email tồn tại thì cập nhật, còn Email không tồn tại thì thêm mới. Khi “Save” thành công thì chương trình vừa hiển thị dữ liệu lên giao diện QListWidget vừa lưu danh sách dữ liệu từ giao diện xuống ổ cứng với định dạng JSON Array. Lưu ý rằng tương ứng với giới tính mà Icon hiển thị của mỗi QListWidgetItem sẽ có Icon khác nhau.
- (5) Chức năng “Delete”: Chương trình sẽ kiểm tra tất cả các dòng Item nào được checked thì sẽ xác nhận xóa, nếu đồng ý xóa thì chương trình sẽ xóa.
- (6) Chức năng “Close”: Chức năng này dùng để đóng cửa sổ phần mềm, khi nhấn vào nút lệnh này chương trình sẽ hiển thị cửa sổ xác nhận Đóng hay không, nếu đồng ý đóng thì chương trình sẽ đóng cửa sổ.
- (7) Chức năng “Selection”: Khi người sử dụng chọn dòng Item nào trong QListWidget thì thông tin chi tiết trong dòng đang chọn sẽ được hiển thị ngược lại lên giao diện.

Hướng dẫn:

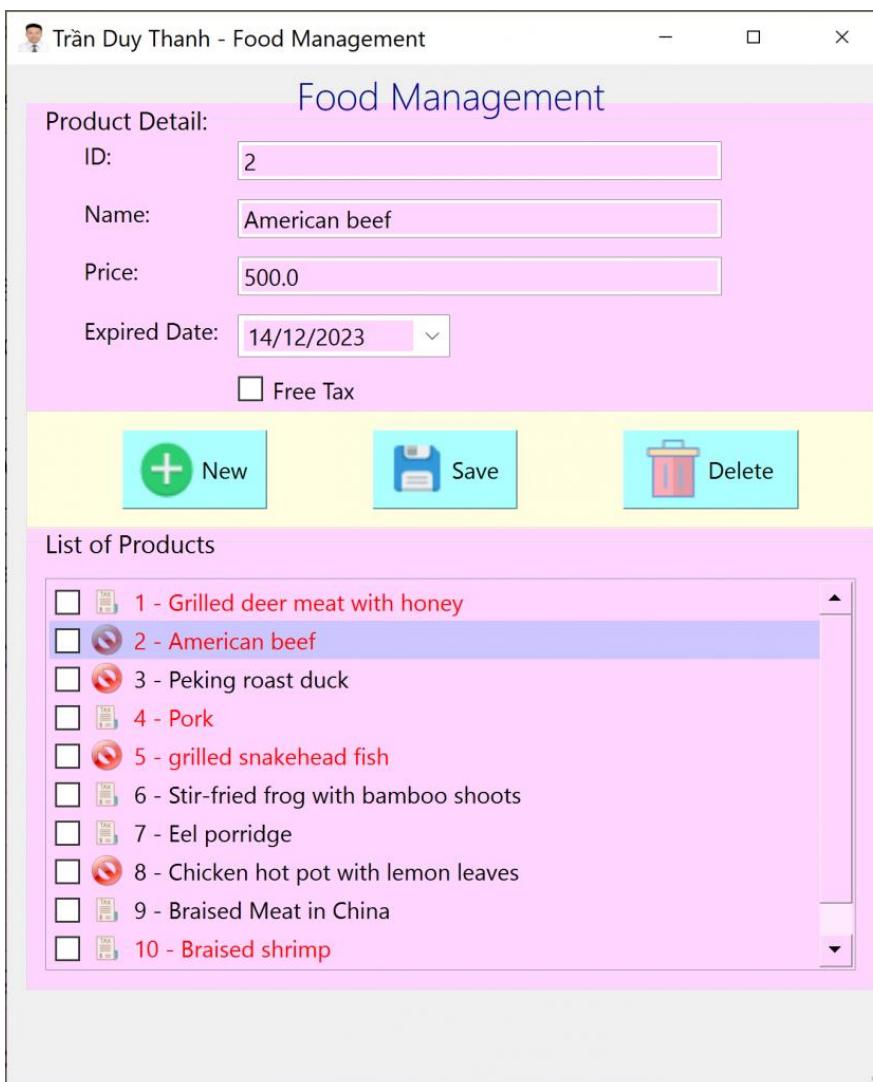
Sinh viên xem hướng dẫn chi tiết tại bài giảng:

<https://tranduythanh.com/2023/12/02/bai-18-qlistwidget-part-2-basic-widgets-pyqt6/>

Câu 101: Phần mềm Food Management - QDateEdit

Yêu cầu:

Thiết kế phần mềm Food Management. Chúng ta sẽ ứng dụng QDateTimeEdit, QCheckBox và QListWidget để viết phần mềm Quản lý thực phẩm. Phần mềm này sẽ cung cấp giao diện nhập dữ liệu Thực phẩm (Sản phẩm) bao gồm: Mã sản phẩm, tên sản phẩm, đơn giá, hạn sử dụng; Sản phẩm thì có sản phẩm được miễn thuế hoặc không, dùng Icon để phân biệt 2 loại sản phẩm này. Đồng thời nếu hạn sử dụng mà trong khoảng 1 tuần tính tới thời điểm hiện tại thì tô màu đỏ cho sản phẩm này:



- Chương trình cung cấp chức năng “Save” Product có 2 nhiệm vụ: Thêm mới và Chính sửa dữ liệu, nếu Product thêm vào mà trùng Id thì chương trình sẽ chỉnh sửa dữ liệu. Thao tác Save sẽ đồng thời lưu toàn bộ Product xuống ổ cứng với định dạng JSONArray (Sinh viên có thể lựa chọn kiểu tập tin bất kỳ đã học)
- Ngoài ra, nút “Delete” sẽ xác nhận xóa các Product được Checked trong QListWidget.

Hướng dẫn:

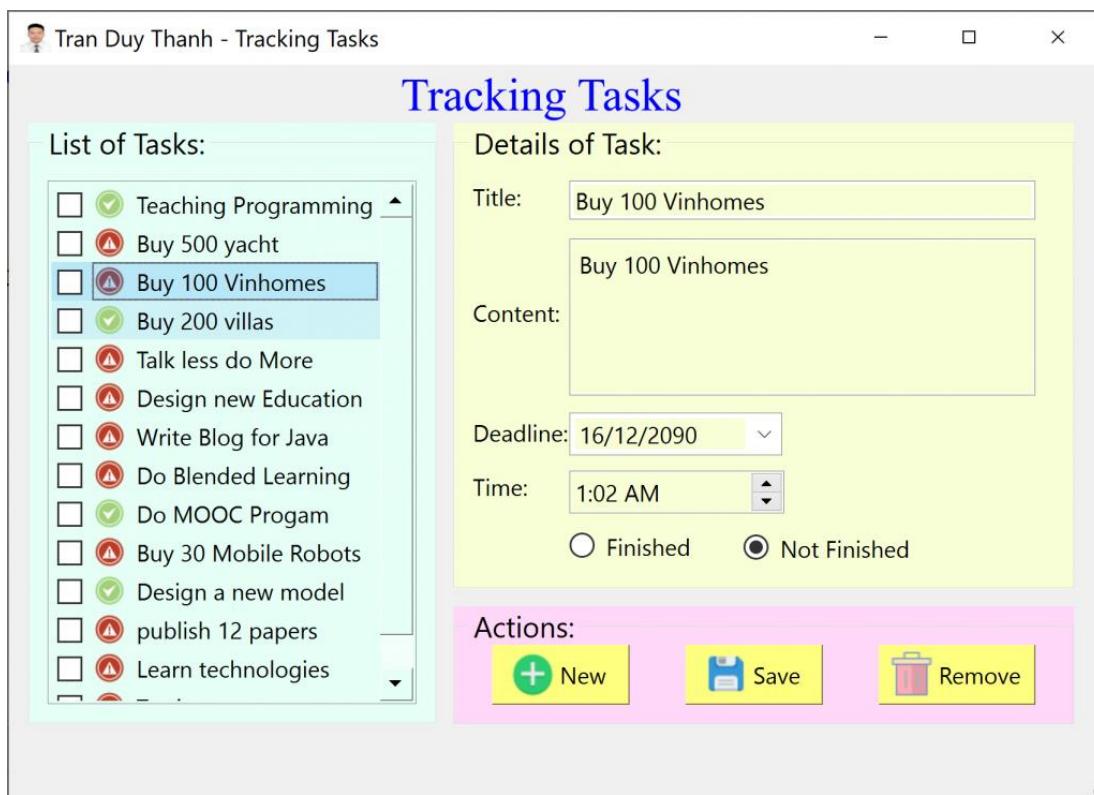
Sinh viên xem hướng dẫn chi tiết tại bài giảng:

<https://tranduythanh.com/2023/12/07/bai-19-qdateedit-basic-widgets-pyqt6/>

Câu 102: Phần mềm Tracking Tasks - QTimeEdit

Yêu cầu:

Áp dụng QDateEdit và QTimeEdit, QListWidget và các widgets cơ bản để phát triển phần mềm Tracking Tasks như dưới đây:



Mô tả các chức năng của phần mềm:

- (1) Chương trình cung cấp giao diện để nhập các Task, thông tin chi tiết của Task bao gồm: Tiêu đề, nội dung thực hiện, ngày hết hạn, giờ hết hạn, và trạng thái đã hoàn thành hay chưa.
- (2) Khi nhập Task thành công thì các Task sẽ được đưa vào QListWidget đồng thời các Task nào chưa hoàn thành thì có biểu tượng màu đỏ, task nào đã hoàn thành thì có biểu tượng màu xanh
- (3) Chương trình sẽ lưu và phục hồi dữ liệu bằng serialize và deserialize JSON ARRAY
- (4) Chương trình cung cấp chức năng chỉnh sửa Task
- (5) Chương trình cung cấp nút xóa, sẽ xóa những Task được checked trong QListWidget
- (6) Chương trình cung cấp chức năng selection trong QListWidget, khi người dùng bấm chuột hoặc di chuyển item trong QListWidget thì chi tiết của Task đang lựa chọn sẽ được hiển thị lại trong các widget bên màn hình bên phải.

Hướng dẫn:

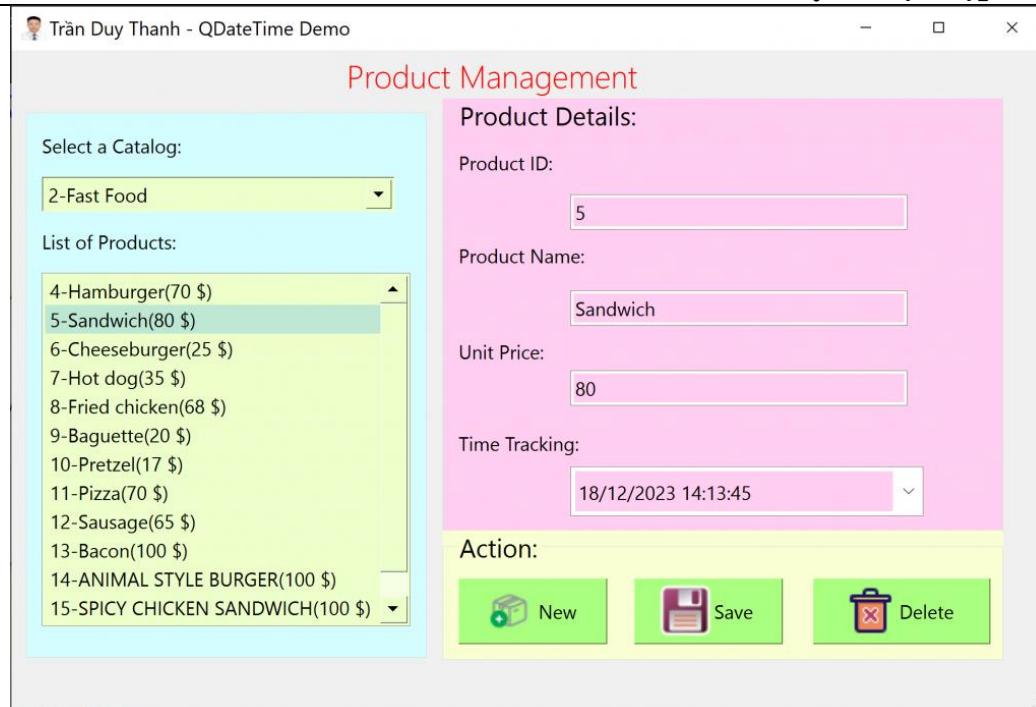
Sinh viên xem hướng dẫn chi tiết tại bài giảng:

<https://tranduythanh.com/2023/12/15/bai-20-qtimeedit-basic-widgets-pyqt6/>

Câu 103: Phần mềm Product Management - QDateTimeEdit

Yêu cầu:

Sinh viên hãy phát triển phần mềm Product Management, phần mềm sử dụng QDateTimeEdit (là một widget kết hợp giữa QDateEdit và QTimeEdit), kết hợp QCombobox, QListWidget và các widgets cơ bản khác. Sinh viên cần thiết kế hướng đối tượng cho các thành phần Danh mục, sản phẩm và có chức năng lưu – đọc tập tin. Giao diện của phần mềm được minh họa như dưới đây:



Mô tả các chức năng của phần mềm:

- (1) Chương trình gồm có DANH MỤC và SẢN PHẨM. 1 Danh Mục có nhiều Sản phẩm, và một sản phẩm chỉ thuộc về một danh mục.
- (2) Thông tin chi tiết của một Danh Mục bao gồm: Mã Danh Mục, Tên Danh Mục
- (3) Thông tin chi tiết của một Sản Phẩm bao gồm: Mã sản phẩm, tên sản phẩm, giá, thời gian tracking (cứ hiểu đại là thời gian muốn theo làm mốc theo dõi, bạn có thể tạo đại một thuộc tính nào đó chứa ngày tháng năm giờ phút giây).
- (4) Chương trình sẽ nạp dữ liệu có sẵn trong “database.json”, mô hình hóa vào các lớp đối tượng Dataset (là đối tượng chứa nhiều Category), Category (Là đối tượng chứa nhiều Product) và Product.
- (5) Danh sách các Category sẽ được nạp vào QCombobox.
- (6) Khi chọn Category nào trong QCombobox thì danh sách Product của Category này sẽ được hiển thị vào QListWidget
- (7) Khi chọn Product trong QListWidget thì thông tin chi tiết của nó sẽ được hiển thị vào nhóm chi tiết bên màn hình “Product Details”.
- (8) Nút “New” sẽ xóa trắng các ô nhập liệu và focus tới ô ID.

- (9) Nút “Save” sẽ lưu Product vào các selected Category tương ứng. Đặc biệt nút “Save” sẽ xử lý 2 tác vụ: Lưu thêm mới Product và lưu Cập nhật Product. Đồng thời cập nhật lại cơ sở dữ liệu
- (10) Nút “Delete” sẽ xóa Product đang chọn, đồng thời cập nhật lại cơ sở dữ liệu.

Hướng dẫn:

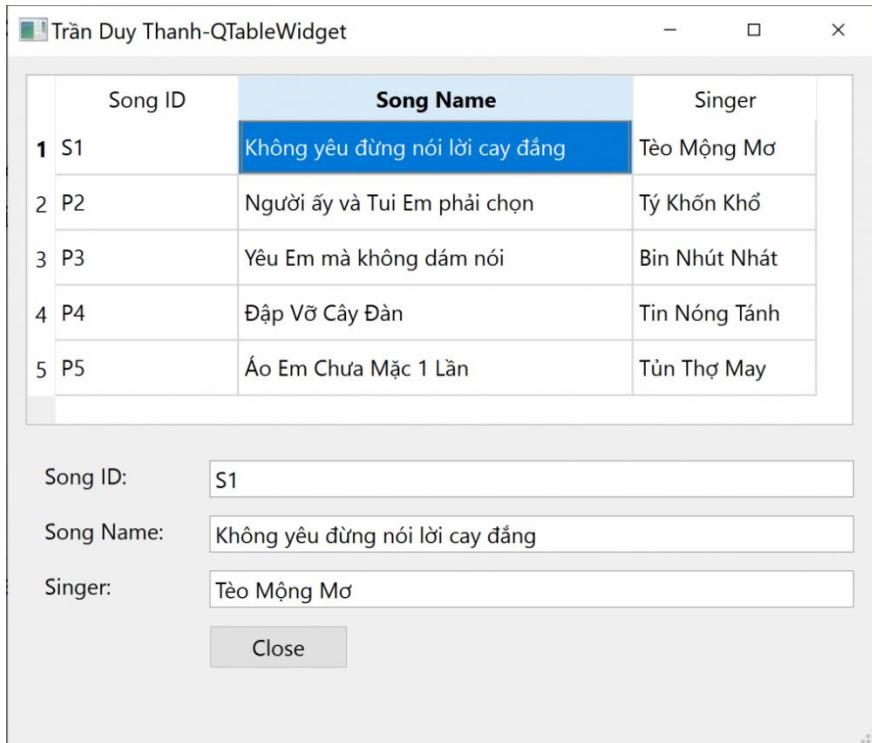
Sinh viên xem hướng dẫn chi tiết tại bài giảng:

<https://tranduythanh.com/2023/12/29/bai-21-qdatetimeedit-basic-widgets-pyqt6/>

Câu 104: Phần mềm Quản lý bài hát - QTableWidgetItem - Design Time

Yêu cầu:

Sử dụng QTableWidgetItem để hiển thị dữ liệu dạng bảng và xử lý sự kiện lựa chọn trên mỗi dòng của QTableWidgetItem như dưới đây:



Sinh viên kéo thả QTableWidgetItem, tạo các cột, dòng dữ liệu bằng Qt Designer, sau đó dùng signal **itemSelectionChanged** để viết sự kiện lựa chọn trên mỗi dòng của QTableWidgetItem để hiển thị chi tiết thông tin bài hát và các Widget QLineEdit.

Hướng dẫn:

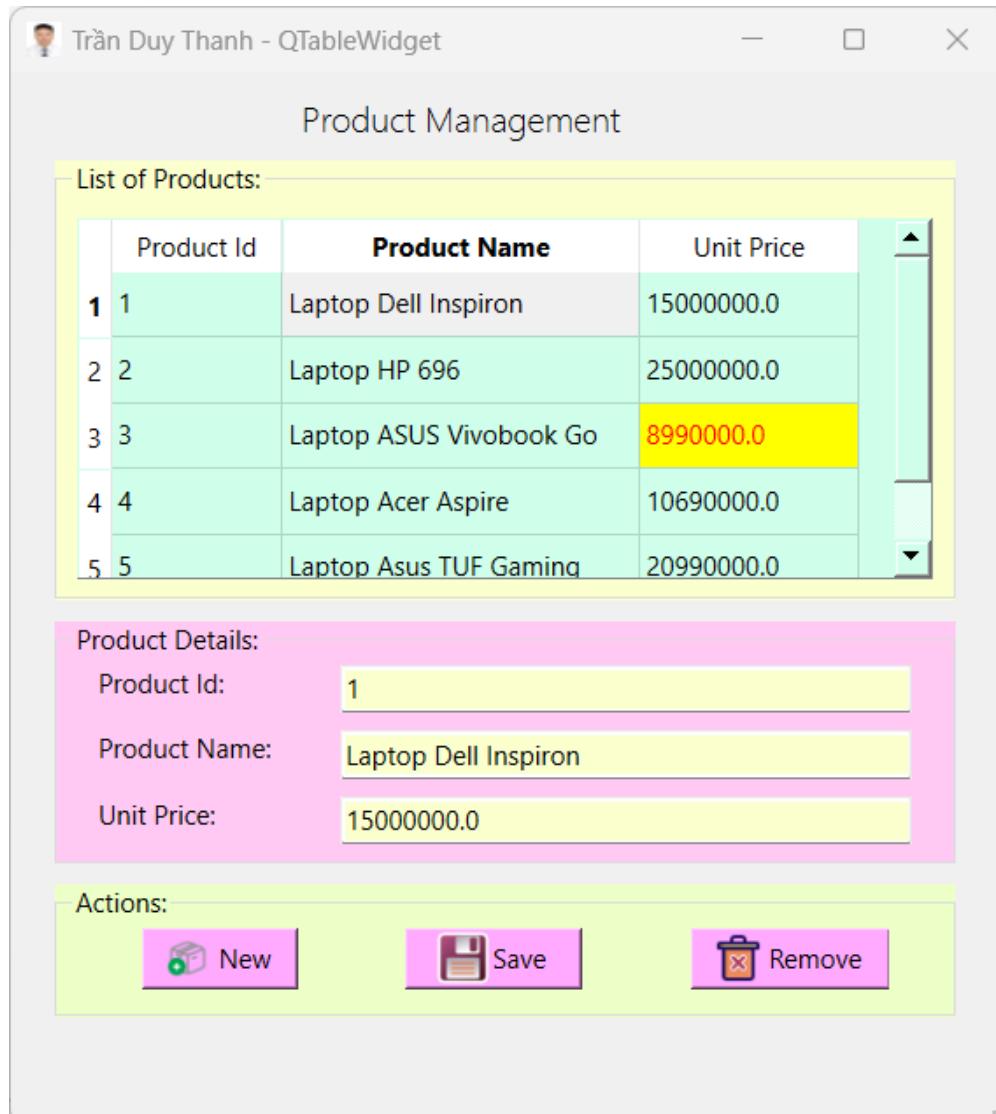
Sinh viên xem hướng dẫn chi tiết tại bài giảng:

<https://tranduythanh.com/2023/12/30/bai-22-hien-thi-du-lieu-dang-bang-qtablewidget-part-1/>

Câu 105: Phần mềm Product Management – QTableWidgetItem runtime

Yêu cầu:

Thiết kế phần mềm Product management, sử dụng QTableWidgetItem và kết hợp với các widget cơ bản khác cùng với mô hình hướng đối tượng, xử lý tập tin. Giao diện minh họa phần mềm như dưới đây:



Sinh viên cần cung cấp các chức năng yêu cầu chính sau:

- (1) Mô hình hóa hướng đối tượng trong Python
- (2) Chức năng thêm, sửa, xóa dữ liệu
- (3) Chức năng hiển thị dữ liệu, những Sản phẩm nào có giá <10 triệu thì tô nền vàng chữ đỏ.
- (4) Chức năng lưu dữ liệu từ QTableWidgetItem xuống JSON (hoặc tập tin bất kỳ đã được học trong môn học)
- (5) Chức năng đọc dữ liệu từ JSON lên QTableWidgetItem (hoặc tập tin bất kỳ đã được học trong môn học)
- (6) Chức năng xử lý sự kiện người dùng chọn các Item trên QTableWidgetItem
- (7) Cũng như cách dùng QMessageBox để điều hướng các lựa chọn của người dùng.

Hướng dẫn:

Sinh viên xem hướng dẫn chi tiết tại bài giảng:

<https://tranduythanh.com/2024/01/01/bai-23-xu-ly-du-lieu-dang-bang-qtablewidget-part-2/>

Câu 106: Phần mềm Product Management – Pickle File

Yêu cầu:

Viết chương trình quản lý sản phẩm của công ty **Đại Phúc** với các tính năng cho phép hiển thị danh sách Sản phẩm, thêm, xóa và cập nhật thông tin sản phẩm.

Thông tin mỗi sản phẩm bao gồm: Mã sản phẩm, tên sản phẩm, hãng sản xuất, giá thành.

Lưu ý chương trình viết theo dạng Module và hướng đối tượng

Sử dụng **Pickle** File để lưu và đọc dữ liệu.



Hướng dẫn:

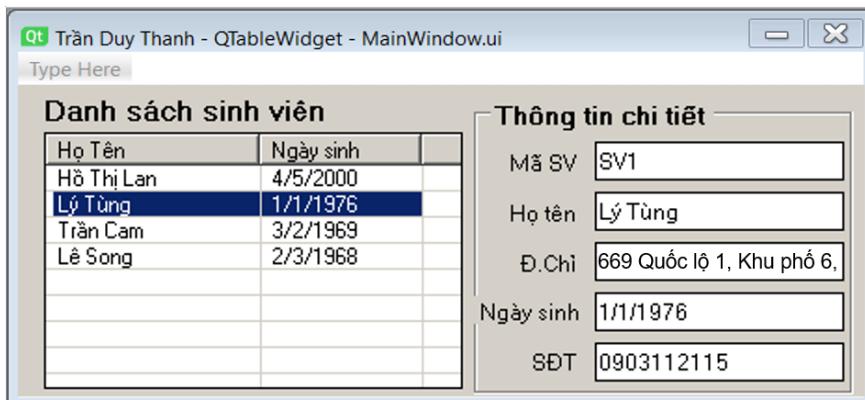
Sinh viên có thể sử dụng Tkinter hoặc PyQt6 để thực hiện phần mềm này.

Câu 107: Phần mềm Student Management – Excel File

Yêu cầu:

Tạo một file Excel chứa danh sách Sinh viên có 5 cột: Mã Sinh viên, họ tên, địa chỉ, Ngày sinh, số điện thoại. Nhập khoảng 4 Sinh viên vào file Excel này.

Sau đó viết mô hình hướng đối tượng, và nạp dữ liệu từ file Excel này lên giao diện như dưới đây:



(1) Danh sách Sinh viên được nạp vào QTableWidgetItem

(2) Khi nhấn vào Sinh viên nào trong QTableWidgetItem thì thông tin chi tiết được hiển thị vào các ô QLineEdit.

Hướng dẫn:

Sinh viên xem lại kiến thức được trình bày trong môn học và tìm hiểu thêm trên mạng để thực hiện bài làm.

Câu 108: Phần mềm Music Management – XML File

Yêu cầu:

Viết phần mềm Music Management, yêu cầu Sinh viên thực hiện:

(1) Thiết kế 2 file XML:

+File “singer.xml”: Gồm tập hợp các dữ liệu là Ca Sĩ, thông tin mỗi Ca Sĩ gồm có: Mã Ca Sĩ, Tên Ca Sĩ

+File “song.xml”: Gồm tập hợp các dữ liệu là Bài Hát, thông tin mỗi Bài Hát gồm có: Mã bài hát, Tên bài hát, Mã Ca Sĩ, Tên Tác giả sáng tác.

Hai file XML này, Sinh viên phải tự thiết kế và nhập dữ liệu sao cho phù hợp với yêu cầu: Một Ca sĩ có thể hát nhiều bài hát, và Một Bài hát có thể được hát bởi nhiều Ca Sĩ.

(2) Sinh viên cần lập trình hướng đối tượng và thiết kế giao diện như dưới đây:



(1) Danh sách Ca sĩ sẽ được nạp vào QCombobox

(2) Khi chọn Ca sĩ nào trong QCombobox thì danh sách Bài hát của Ca sĩ sẽ được nạp vào QTableWidget

Hướng dẫn:

Sinh viên xem lại kiến thức được trình bày trong môn học và tìm hiểu thêm trên mạng để thực hiện bài làm.

Câu 109: Phần mềm Electronic Components Management – JSON File**Yêu cầu:**

Viết phần mềm Electronic Components Management, yêu cầu Sinh viên thực hiện:

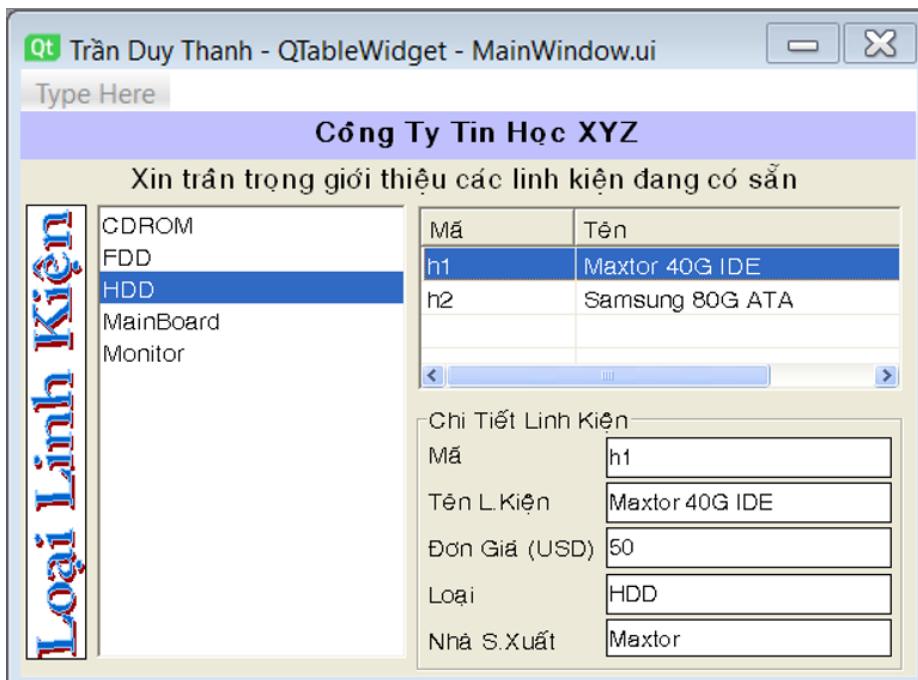
(1) Thiết kế 2 file JSON:

+File “category.json”: Gồm tập hợp các dữ liệu là các danh mục linh kiện Như CDROOM, HDD, SSD, MainBoard, Monitor...

+File “product.json”: Gồm tập hợp các dữ liệu là các linh kiện được phân loại theo category. Thông tin chi tiết từng link kiện gồm Mã linh kiện, tên linh kiện, đơn giá, nhà sản xuất

Hai file JSON này, Sinh viên phải tự thiết kế và nhập dữ liệu sao cho phù hợp với yêu cầu.

(2) Sinh viên cần lập trình hướng đối tượng và thiết kế giao diện như dưới đây:



- (1) Danh sách Danh mục linh kiện sẽ được nạp vào QListWidget
- (2) Khi chọn Danh mục linh kiện nào trong QListWidget thì danh sách các linh kiện của danh mục này sẽ được nạp vào vào QTableWidget
- (3) Khi chọn Linh kiện nào trong QTableWidget thì thông tin chi tiết của linh kiện sẽ được nạp vào các QLineEdit ở mục chi tiết linh kiện.

Hướng dẫn:

Sinh viên xem lại kiến thức được trình bày trong môn học và tìm hiểu thêm trên mạng để thực hiện bài làm.

Câu 110: Phần mềm Employee Management – CSV File

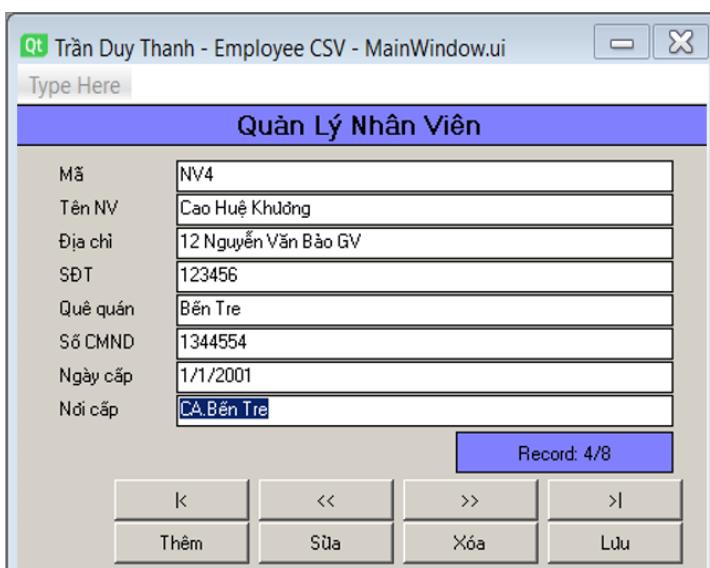
Yêu cầu:

Viết phần mềm Employee Management, yêu cầu Sinh viên thực hiện:

- (1) Thiết kế 1 file CSV:

+File “employee.csv”: Gồm tập hợp các dữ liệu là Employee gồm các thông tin như Mã nhân viên, tên nhân viên, địa chỉ, số điện thoại, quê quán Số CMND/CCCD, Ngày cấp, Nơi cấp. File CSV này, Sinh viên phải tự thiết kế và nhập dữ liệu sao cho phù hợp với yêu cầu (khoảng 8 tới 10 dữ liệu Nhân viên mẫu)

- (2) Sinh viên cần lập trình hướng đối tượng và thiết kế giao diện như dưới đây:



- (1) Danh sách Nhân viên sẽ được nạp vào collection, và Nhân viên đầu tiên trong danh sách sẽ hiển thị lên các QLineEdit. Và mặc định các QLineEdit là readonly
- (2) “<” nút duyệt và hiển thị nhân viên đầu tiên
- (3) “>” nút duyệt và hiển thị nhân viên cuối cùng
- (4) “<<” nút duyệt và hiển thị nhân viên phía trước
- (5) “>>” nút duyệt và hiển thị nhân viên phía sau

Chú ý các thao tác duyệt phải hiển thị số thứ tự nhân viên hiện tại / tổng nhân viên

- (6) “Thêm”: Xóa trống dữ liệu trên các QLineEdit, và focus vào ô Mã. Tất cả các QLineEdit là editable
- (7) “Sửa”: Tất cả các QLineEdit là editable. Dữ liệu trên QLineEdit giữ nguyên
- (8) “Xóa”: Xóa Nhân viên hiện tại khỏi bộ nhớ
- (9) “Lưu”: Dữ liệu sẽ được cập nhật xuống file CSV.

Hướng dẫn:

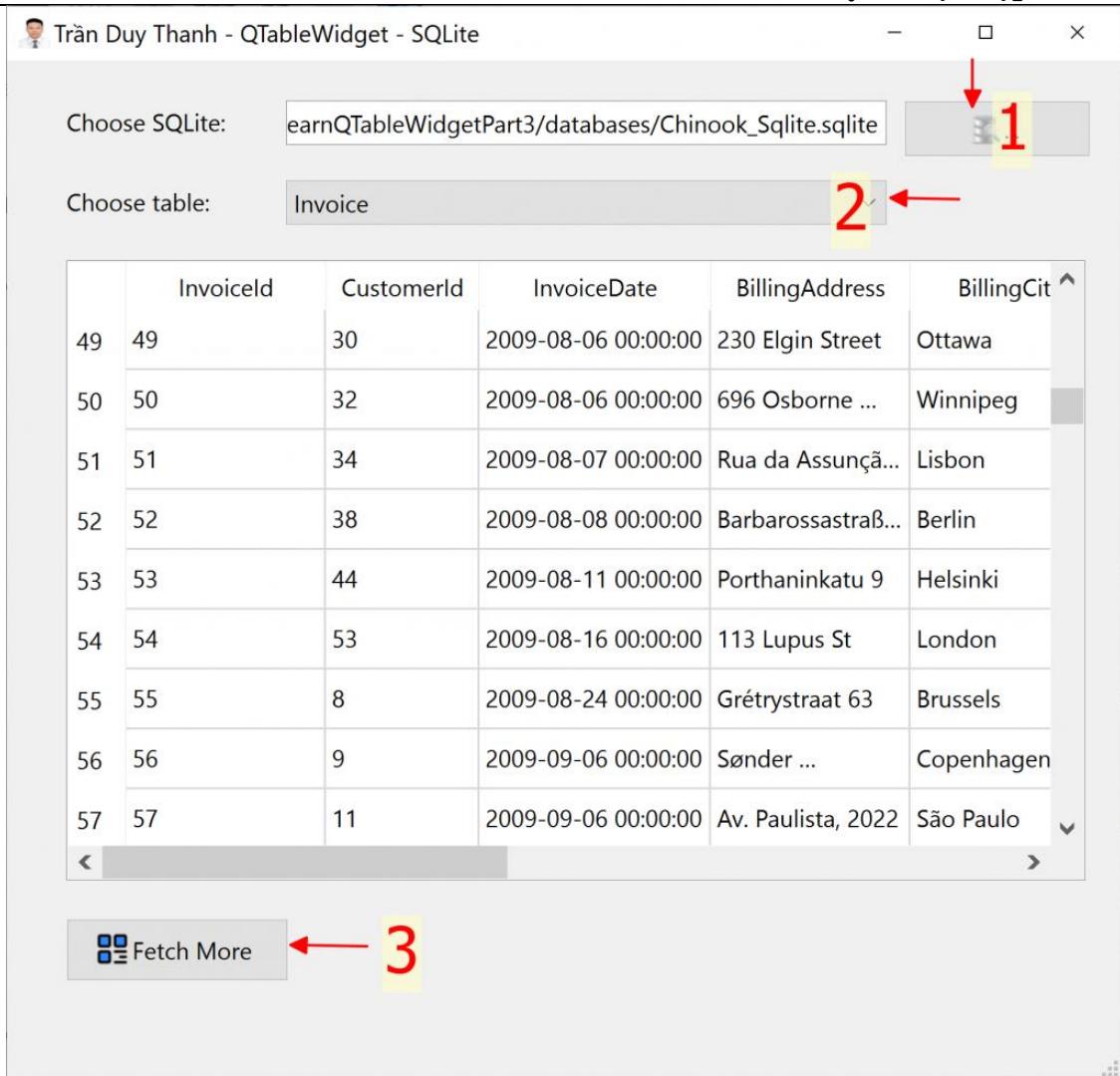
Sinh viên xem lại kiến thức được trình bày trong môn học và tìm hiểu thêm trên mạng để thực hiện bài làm.

Câu 111: Chinook SQLite Database – QTableWidgetItem()**

Yêu cầu:

(Bài này Sinh viên cần bổ sung thêm kiến thức về Cơ sở dữ liệu/Hệ quản trị cơ sở dữ liệu).

Sinh viên làm quen với SQLite Database để lưu trữ và xử lý dữ liệu, Bài tập này Sinh viên cần thực hiện một số kỹ thuật để kết nối và truy vấn dữ liệu từ SQLite lên QTableWidgetItem. Sinh viên viết các mã lệnh để tự động kết nối, đọc danh sách các bảng trong cơ sở dữ liệu và truy vấn danh sách dữ liệu lên giao diện QTableWidgetItem:



Các chức năng chính của phần mềm bao gồm:

- (1) Cho người dùng lựa chọn một cơ sở dữ liệu SQLite bất kỳ để kết nối
- (2) Chương trình sẽ tự động đọc danh sách các bảng dữ liệu nằm bên trong SQLite
- (3) Người dùng chọn bảng dữ liệu nào thì chương trình sẽ truy vấn các dữ liệu ở bên trong bảng này lên giao diện QTableWidget
- (4) Cung cấp chức năng Fetch More để tiếp tục đọc các dữ liệu trong bảng trong trường hợp bảng có nhiều dòng dữ liệu (Ví dụ lớn hơn 256 dòng dữ liệu)

Hướng dẫn:

Sinh viên xem hướng dẫn chi tiết tại bài giảng:

<https://tranduythanh.com/2024/01/02/bai-24-xu-ly-du-lieu-dang-bang-qtablewidget-va-sqlite-database-part-3/>

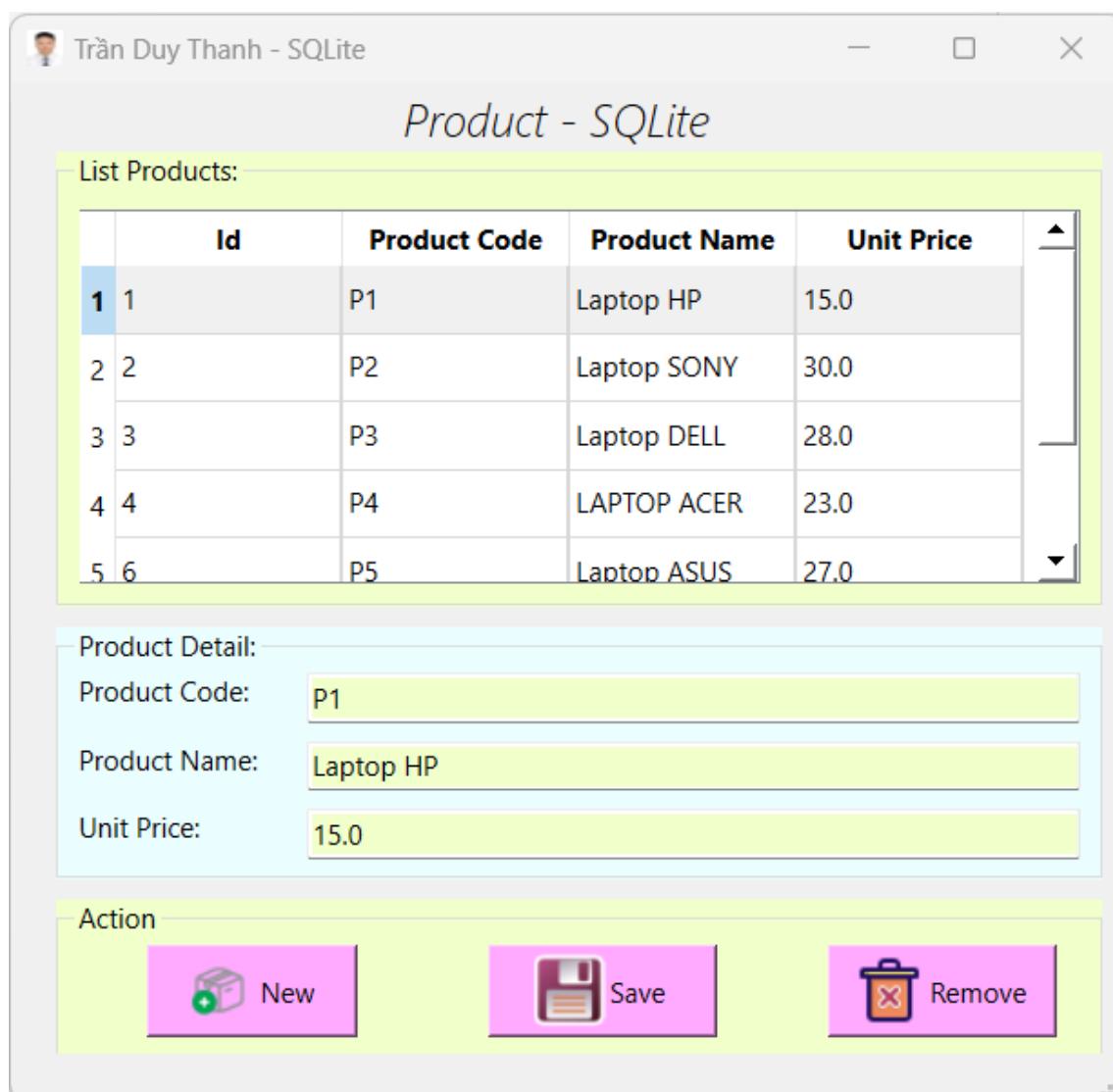
Câu 112: Phần mềm Product Management – SQLite Database(**)

Yêu cầu:

(Bài này Sinh viên cần bổ sung thêm kiến thức về Cơ sở dữ liệu/Hệ quản trị cơ sở dữ liệu).

Thiết kế phần mềm quản lý Product, sử dụng SQLite Database và QTableWidget.

Giao diện minh họa như dưới đây:



Sinh viên thực hiện các yêu cầu sau:

- (1) Tạo Cơ sở dữ liệu SQLite bằng DB Browser, thiết kế các bảng dữ liệu có cột Id Primary Key là Auto Increment.

-
- (2) Viết chức năng đọc toàn bộ dữ liệu trong CSDL SQLite lên giao diện QTableWidget trong mục List Products
 - (3) Xử lý sự kiện người dùng chọn các dòng dữ liệu trên QTableWidget và hiển thị thông tin chi tiết xuống phần Product Details
 - (4) Chức năng “New” sẽ xóa dữ liệu đang nhập ở các ô QLineEdit và focus tới ô Product code
 - (5) Chức năng “Save”, viết mã lệnh để chương trình tự xử lý 2 trường hợp là lưu mới dữ liệu xuống SQLite hoặc là lưu cập nhật xuống SQLite
 - (6) Chức năng “Remove”, viết mã lệnh để xóa dòng dữ liệu đang chọn, cho người dùng xác thực trước khi xóa.

Hướng dẫn:

Sinh viên xem hướng dẫn chi tiết tại bài giảng:

<https://tranduythanh.com/2024/01/04/bai-25-xu-ly-du-lieu-dang-bang-qtablewidget-va-sqlite-database-part-4/>

MODULE 6

THƯ VIỆN TRONG PYTHON

Nội dung kiến thức thực hành:

- Các khái niệm liên quan tới Thư viện trong lập trình
- Cách thức tự định nghĩa thư viện.
- Xử lý Một số thư viện thông dụng trong Python
 - + Xử lý dữ liệu mảng với thư viện Numpy.
 - + Tổ chức xử lý dữ liệu bảng với thư viện Pandas.
 - + Trực quan hóa dữ liệu với thư viện Matplotlib & Seaborn.
 - + Làm việc với Random module.
 - + Cách Tích hợp Matplotlib & Seaborn vào giao diện PyQt6 – Qt Designer
 - + Cách tương tác hướng đối tượng với các thư viện
 - + Cách đóng gói và xuất bản phần mềm

Câu 113: Phân biệt Module, Package, Library

Yêu cầu:

Sinh viên hãy phân biệt các khái niệm dưới đây và cho ví dụ minh họa:

-Module

-Package

-Library

Hướng dẫn:

Sinh viên xem lại kiến thức được trình bày trong môn học và tìm hiểu thêm trên mạng để thực hiện bài làm.

Câu 114: Trình bày cách cài đặt và sử dụng thư viện Numpy**Yêu cầu:**

Sinh viên hãy trình bày cách cài đặt và sử dụng thư viện Numpy, liệt kê các đặc điểm chính của Numpy. Cho một vài ví dụ liên quan tới sử dụng Numpy, chẳng hạn như xử lý mảng 1 chiều, xử lý mảng nhiều chiều.

Hướng dẫn:

Sinh viên xem lại kiến thức được trình bày trong môn học và tìm hiểu thêm trên mạng để thực hiện bài làm.

Câu 115: Một số câu trắc nghiệm cơ bản liên quan tới thư viện Numpy**Yêu cầu:**

Dưới đây là một số câu hỏi trắc nghiệm liên quan tới thư viện Numpy để ôn tập:

1. **Câu hỏi:** Để import thư viện NumPy trong Python, bạn sử dụng câu lệnh nào sau đây?
 - A. import numpy as np
 - B. from numpy import *
 - C. include numpy
 - D. require numpy
2. **Câu hỏi:** Để tạo một mảng NumPy 1 chiều từ một danh sách Python, bạn sử dụng hàm nào?
 - A. np.matrix()
 - B. np.ndarray()
 - C. np.array()
 - D. np.arraylist()
3. **Câu hỏi:** Hàm np.zeros((3, 4)) sẽ tạo ra một mảng NumPy với các giá trị như thế nào?
 - A. Mảng 2 chiều có kích thước (3, 4) chứa toàn giá trị 1.
 - B. Mảng 2 chiều có kích thước (3, 4) chứa toàn giá trị 0.
 - C. Mảng 3 chiều có kích thước (4, 3) chứa toàn giá trị 0.
 - D. Mảng 3 chiều có kích thước (3, 4, 0).
4. **Câu hỏi:** Hàm np.eye(3) tạo ra một mảng NumPy như thế nào?
 - A. Mảng vuông 3x3 với đường chéo chính có giá trị 1 và các phần tử khác có giá trị 0.
 - B. Mảng vuông 3x3 với tất cả các phần tử có giá trị 1.
 - C. Mảng 2x3 với tất cả các phần tử có giá trị 0.
 - D. Mảng 3x2 với tất cả các phần tử có giá trị 0.
5. **Câu hỏi:** Để tính tổng các phần tử trong mảng NumPy arr, bạn sử dụng hàm nào sau đây?
 - A. np.add(arr)
 - B. np.sum(arr)
 - C. np.total(arr)
 - D. np.aggregate(arr)
6. **Câu hỏi:** Hàm np.linspace(0, 10, 5) sẽ trả về một mảng NumPy chứa các giá trị như thế nào?

- A. Mảng gồm 10 phần tử từ 0 đến 10 với khoảng cách đều nhau.
 - B. Mảng gồm 10 phần tử từ 0 đến 5 với khoảng cách đều nhau.
 - C. Mảng gồm 5 phần tử từ 0 đến 5 với khoảng cách đều nhau.
 - D. Mảng gồm 5 phần tử từ 0 đến 10 với khoảng cách đều nhau.
7. **Câu hỏi:** Để thực hiện phép nhân ma trận trong NumPy, bạn sử dụng hàm nào sau đây?
- A. np.multiply()
 - B. np.cross()
 - C. np.dot()
 - D. np.product()
8. **Câu hỏi:** Hàm np.random.rand(4, 3) tạo ra một mảng NumPy với các giá trị như thế nào?
- A. Mảng 4x3 chứa các giá trị ngẫu nhiên từ phân phối chuẩn.
 - B. Mảng 4x3 chứa các giá trị ngẫu nhiên từ phân phối đều từ 0 đến 1.
 - C. Mảng 3x4 chứa các giá trị ngẫu nhiên từ phân phối chuẩn.
 - D. Mảng 3x4 chứa các giá trị ngẫu nhiên từ phân phối đều từ 0 đến 1.
9. **Câu hỏi:** Để chuyển vị một ma trận trong NumPy, bạn sử dụng phương thức nào?
- A. reverse()
 - B. flip()
 - C. swapaxes()
 - D. transpose()
10. **Câu hỏi:** Hàm np.argmax(arr) trả về gì?
- A. Giá trị lớn nhất trong mảng arr.
 - B. Số lượng giá trị lớn hơn 0 trong mảng arr.
 - C. Chỉ số của giá trị lớn nhất trong mảng arr.
 - D. Số lượng giá trị bằng 0 trong mảng arr.

Hướng dẫn:

Sinh viên xem lại kiến thức được trình bày trong môn học và tìm hiểu thêm trên mạng để thực hiện bài làm.

Câu 116: Các phép toán trên Mảng 1 chiều sử dụng Numpy

Yêu cầu:

Sinh viên sử dụng Numpy để tạo ra mảng ngẫu nhiên có 10 phần tử có các giá trị từ [-100 tới 500]. Sau đó thực hiện các chức năng:

- (1) Xuất toàn bộ giá trị trong mảng
- (2) Xuất các phần tử có vị trí từ 2 tới 5
- (3) Xuất các phần tử có giá trị âm
- (4) Xuất các phần tử có giá trị từ x tới y. Với x, y là các số nhập từ bàn phím
- (5) Lọc ra các số âm trong mảng
- (6) Sắp xếp mảng tăng dần

(7) Sắp xếp mảng giảm dần

(8) Xuất các giá trị thống kê cơ bản: Giá trị min, giá trị max, giá trị trung bình, giá trị trung vị, độ lệch chuẩn

(9) Xóa các phần tử là các số chính phương khỏi mảng

(10) Chèn X vào vị trí V, với X và V là giá trị nhập từ bàn phím

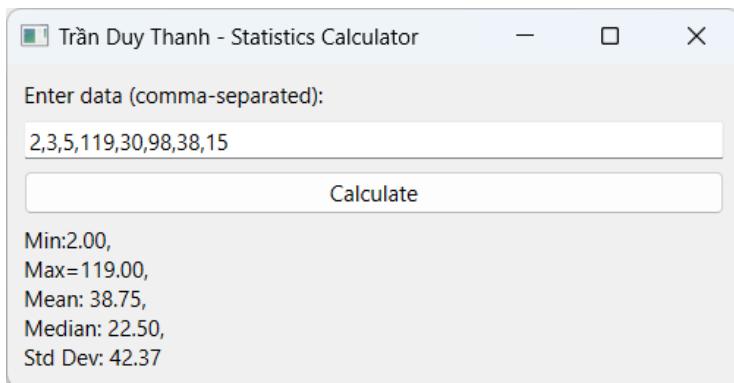
Hướng dẫn:

Sinh viên xem lại kiến thức được trình bày trong môn học và tìm hiểu thêm trên mạng để thực hiện bài làm.

Câu 117: Thống kê cơ bản trên Mảng 1 chiều sử dụng Numpy - GUI

Yêu cầu:

Sinh viên thiết kế giao diện PyQt6 như hình dưới đây:



(1) Người sử dụng nhập các giá trị cách nhau bởi dấu phẩy hoặc dấu chấm phẩy trong ô QLineEdit

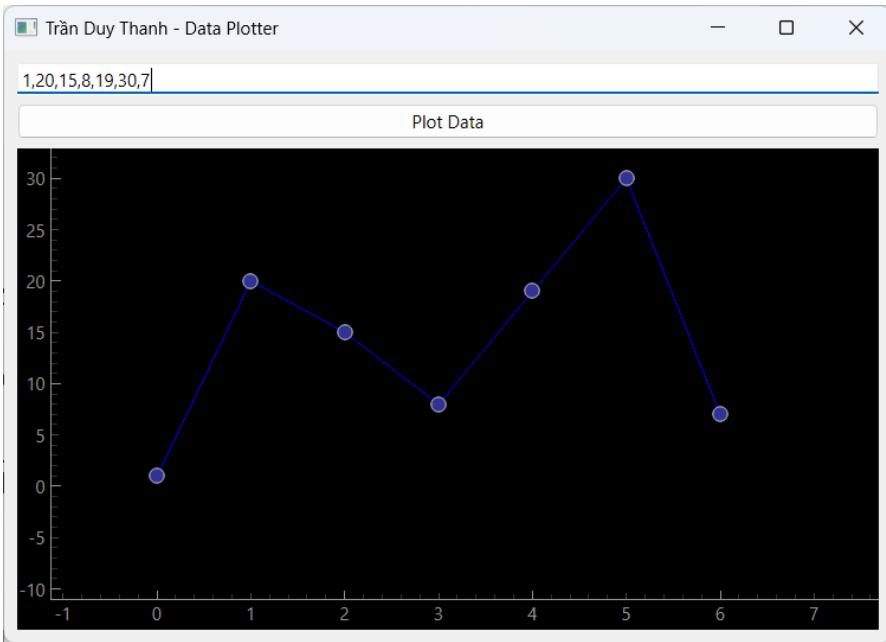
(2) Khi nhấn “Calculate”: Viết mã lệnh để tách lấy dữ liệu từ QLineEdit để phân tích thành mảng 1 chiều sử dụng Numpy. Sau đó xuất các giá trị thống kê cơ bản vào QLabel.

Hướng dẫn:

Sinh viên xem lại kiến thức được trình bày trong môn học và tìm hiểu thêm trên mạng để thực hiện bài làm.

Câu 118: Trực quan hóa Mảng 1 chiều sử dụng Numpy - GUI**Yêu cầu:**

Sinh viên thiết kế giao diện như dưới đây:



- (1) Người sử dụng nhập các giá trị cách nhau bởi dấu phẩy hoặc dấu chấm phẩy trong ô QLineEdit
- (2) Khi nhấn “Plot Data”: Viết mã lệnh để tách lấy dữ liệu từ QLineEdit để phân tích thành mảng 1 chiều sử dụng Numpy. Sau đó sử dụng pyqtgraph để trực quan hóa dữ liệu này.

Hướng dẫn:

Sinh viên xem lại kiến thức được trình bày trong môn học và tìm hiểu thêm trên mạng để thực hiện bài làm.

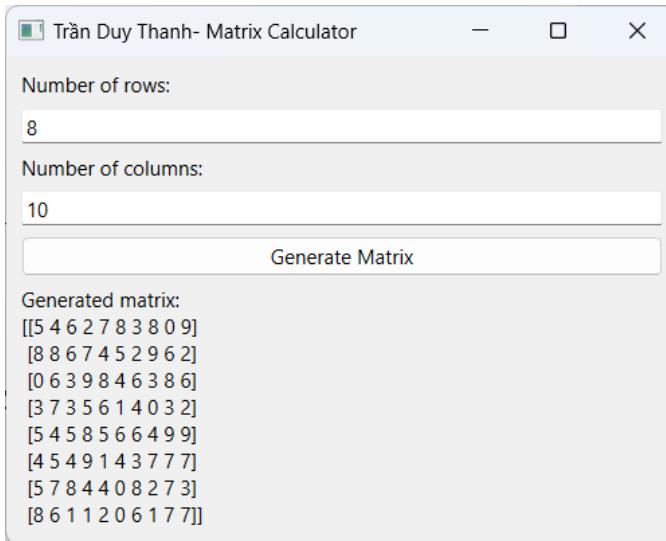
Tham khảo bài giảng:

<https://tranduythanh.com/2024/01/14/bai-31-truc-quan-hoa-du-lieu-pyqtgraph-pyqt6-part-1/>

<https://tranduythanh.com/2024/01/17/bai-32-truc-quan-hoa-du-lieu-pyqtgraph-pyqt6-part-2/>

Câu 119: Tạo ma trận sử dụng Numpy - GUI**Yêu cầu:**

Sinh viên thiết kế giao diện và viết mã lệnh để tạo ma trận như hình dưới đây:



Nhập số dòng và số cột. Sau đó nhấn “Generate Matrix” thì chương trình sẽ xuất ma trận ngẫu nhiên tùy ý lên giao diện.

Hướng dẫn:

Sinh viên xem lại kiến thức được trình bày trong môn học và tìm hiểu thêm trên mạng để thực hiện bài làm.

Câu 120: Các phép toán trên ma trận sử dụng Numpy**Yêu cầu:**

Sinh viên hãy tạo ngẫu nhiên một ma trận có N dòng và M cột, các giá trị ngẫu nhiên từ -100 tới 100. Sau đó thực hiện các chức năng sau:

- (1) Tính định thức ma trận
- (2) Tìm ma trận nghịch đảo
- (3) Sắp xếp ma trận theo hàng

- (4) Sắp xếp ma trận theo cột
- (5) Sắp xếp ma trận theo giá trị trung bình của từng hàng
- (6) Thay đổi giá trị của một phần tử trong ma trận: Nhập dòng và cột của phần tử muốn đổi giá trị, sau đó cho phép nhập giá trị mới từ bàn phím để đổi giá trị.
- (7) Thay đổi giá trị một cột của ma trận theo điều kiện: Nhập vị trí cột muốn đổi giá trị, sau đó tăng giá trị lên 2.
- (8) Cộng thêm một vector vào từng hàng của ma trận: Sinh viên tạo ra một vector phù hợp với ma trận gốc để có thể thực hiện phép cộng được.
- (9) Tính hạng của ma trận
- (10) Tính phân tích Singular Value Decomposition (SVD)

Hướng dẫn:

Sinh viên xem lại kiến thức được trình bày trong môn học và tìm hiểu thêm trên mạng để thực hiện bài làm.

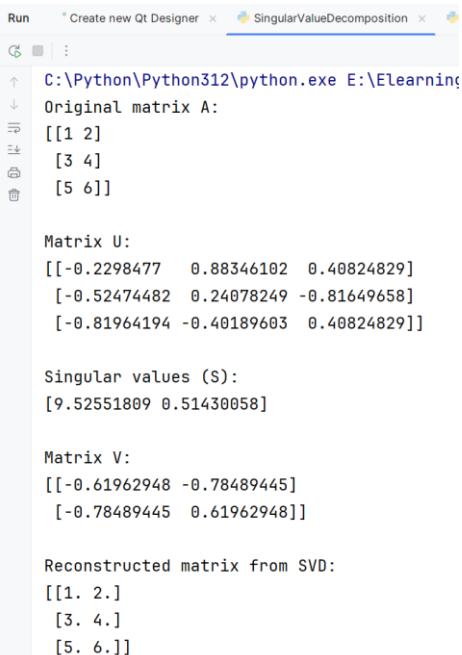
Giải thích về Phân tích Singular Value Decomposition (SVD):

Phân tích Singular Value Decomposition (SVD) là một trong những phương pháp phân tích ma trận quan trọng trong đại số tuyến tính và khoa học dữ liệu. Trong NumPy, ta có thể sử dụng hàm `numpy.linalg.svd()` để thực hiện SVD trên một ma trận. Dưới đây là một ví dụ cụ thể về cách tính và sử dụng SVD bằng NumPy:

```
1 import numpy as np
2 from numpy import zeros
3
4 # Tạo một ma trận đầu vào
5 A = np.array([[1, 2], [3, 4], [5, 6]])
6
7 # Thực hiện Singular Value Decomposition (SVD)
8 U, S, V = np.linalg.svd(A)
9
10 # U là ma trận unitary (orthogonal) cỡ m x m
11 # S là vector chứa các singular values (sắp xếp giảm dần)
12 # V là ma trận unitary (orthogonal) cỡ n x n
13
14 print("Original matrix A:")
15 print(A)
```

```
16 print("\nMatrix U:")
17 print(U)
18 print("\nSingular values (S):")
19 print(S)
20 print("\nMatrix V:")
21 print(V)
22
23 # Tạo lại ma trận ban đầu từ U, S và V
24 Sigma = zeros((A.shape[0], A.shape[1]))
25 # populate Sigma with n x n diagonal matrix
26 Sigma[:A.shape[1], :A.shape[1]] = np.diag(S)
27 # reconstruct matrix
28 A_reconstructed = U.dot(Sigma.dot(V))
29 print("\nReconstructed matrix from SVD:")
30 print(A_reconstructed)
```

Kết quả khi thực hiện đoạn mã trên:



```
C:\Python\Python312\python.exe E:\Elearning\SingularValueDecomposition.py
Original matrix A:
[[1 2]
 [3 4]
 [5 6]]

Matrix U:
[[-0.2298477  0.88346102  0.40824829]
 [-0.52474482  0.24078249 -0.81649658]
 [-0.81964194 -0.40189603  0.40824829]]

Singular values (S):
[9.52551809 0.51430058]

Matrix V:
[[-0.61962948 -0.78489445]
 [-0.78489445  0.61962948]]

Reconstructed matrix from SVD:
[[1. 2.]
 [3. 4.]
 [5. 6.]]
```

Câu 121: Các phép toán trên 2 ma trận sử dụng Numpy

Yêu cầu:

Sinh viên hãy tạo ngẫu nhiên 2 ma trận có N1,N2 dòng và M1,M2 cột, các giá trị ngẫu nhiên từ -100 tới 100. Sau đó thực hiện các chức năng sau:

- (1) Tính tổng 2 ma trận (nếu được)

(2) Tính hiệu 2 ma trận (nếu được)

(3) Tính tích 2 ma trận (nếu được)

(4) Tính tích Hadamard của 2 ma trận (nếu được)

(3) Giải phương trình hệ tuyến tính (nếu được)

Hướng dẫn:

Sinh viên xem lại kiến thức được trình bày trong môn học và tìm hiểu thêm trên mạng để thực hiện bài làm.

Câu 122: Tính Gauss sử dụng Numpy

Yêu cầu:

Phương pháp tính toán Gauss (hoặc phương pháp khử Gauss) được sử dụng để giải hệ phương trình tuyến tính. Trong phương pháp này, ta cố gắng biến đổi ma trận hệ số của hệ phương trình thành ma trận tam giác trên hoặc ma trận đường chéo, sau đó dễ dàng giải hệ phương trình bằng phương pháp lùi.

Để thực hiện phương pháp Gauss sử dụng NumPy, chúng ta có thể áp dụng các phép biến đổi ma trận như phép cộng dòng, nhân dòng với một số vô hướng và hoán vị dòng. Dưới đây là một ví dụ cụ thể:

```
1 import numpy as np
2
3 def gauss_elimination(A, B):
4     """
5         Giải hệ phương trình Ax = B bằng phương pháp Gauss
6         Đầu vào:
7             - A: ma trận hệ số (numpy array)
8             - B: vector cột bên phải (numpy array)
9         Đầu ra:
10            - x: vector nghiệm của hệ phương trình (numpy array)
11            """
12     n = len(B)
13     # Nối ma trận B vào A để tạo thành ma trận mở rộng
14     AB = np.column_stack((A, B))
15 
```

```
16     # Áp dụng phương pháp Gauss để biến đổi ma trận AB thành ma
17     # trận tam giác trên
18     for i in range(n):
19         # Tìm hàng chứa phần tử lớn nhất trong cột i và hoán vị nó
20         # lên đầu cột
21         max_row = np.argmax(np.abs(AB[i:, i])) + i
22         AB[[i, max_row]] = AB[[max_row, i]]
23
24         # Lặp qua các hàng phía dưới để loại bỏ các phần tử trong
25         # cột i
26         for j in range(i + 1, n):
27             ratio = AB[j, i] / AB[i, i]
28             AB[j, :] -= ratio * AB[i, :]
29
30         # Tính nghiệm của hệ phương trình bằng phương pháp lùi
31         x = np.zeros(n)
32         for i in range(n - 1, -1, -1):
33             x[i] = (AB[i, -1] - np.dot(AB[i, :-1], x)) / AB[i, i]
34         return x
35 # Ví dụ sử dụng:
36 A = np.array([[2, 3, 1],
37               [4, 7, 5],
38               [2, 5, 2]], dtype=float)
39 B = np.array([5, 10, 6], dtype=float)
40 x = gauss_elimination(A, B)
41 print("Nghiệm của hệ phương trình là:", x)
```

Trong ví dụ này, hàm **gauss_elimination** thực hiện phương pháp Gauss để giải hệ phương trình tuyến tính $Ax = B$. Các bước chính bao gồm:

- (1) Nối ma trận B vào ma trận A để tạo thành ma trận mở rộng AB.
- (2) Áp dụng Gauss để biến đổi ma trận AB thành ma trận tam giác trên.
- (3) Tính nghiệm của hệ phương trình bằng phương pháp lùi (back substitution).

Đây là một cách tiếp cận cơ bản và hiệu quả để giải hệ phương trình tuyến tính bằng phương pháp Gauss sử dụng NumPy trong Python. Ta có thể thay đổi ma trận A và vector B để thử nghiệm với các bài toán khác nhau.

Hướng dẫn:

Sinh viên xem lại kiến thức được trình bày trong môn học và tìm hiểu thêm trên mạng để thực hiện bài làm.

Câu 123: Trình bày cách cài đặt và sử dụng thư viện Pandas**Yêu cầu:**

Sinh viên hãy trình bày cách cài đặt và sử dụng thư viện Pandas. Cho một vài ví dụ liên quan tới xử lý dữ liệu bảng bằng thư viện Pandas, chẳng hạn như đọc dữ liệu từ CSV bằng Pandas, sắp xếp dữ liệu bằng Pandas...

Hướng dẫn:

Sinh viên xem lại kiến thức được trình bày trong môn học và tìm hiểu thêm trên mạng để thực hiện bài làm.

Câu 124: Phân tích dữ liệu Employee - Pandas**Yêu cầu:**

Cho Dataset Employee: <https://tranduythanh.com/datasets/employee.csv>

Sinh viên sử dụng Pandas và thực hiện các yêu cầu sau:

- (0) Tải dữ liệu về và bổ sung thêm 5 Employee với các dữ liệu khác nhau
- (1) Xuất toàn bộ dữ liệu Employee ra màn hình console
- (2) Lọc ra các Employee có năm sinh 2001
- (3) Xuất TOP 3 Employee có tuổi lớn nhất
- (4) Lọc ra các Employee có role là Tester
- (5) Thống kê số lượng Employee theo từng Role
- (6) Thiết kế giao diện và hiển thị danh sách employee lên QTableWidget

Hướng dẫn:

Sinh viên xem lại kiến thức được trình bày trong môn học và tìm hiểu thêm trên mạng để thực hiện bài làm.

Câu 125: Phân tích dữ liệu Sales - Pandas

Yêu cầu:

Cho Dataset Sales: <https://tranduythanh.com/datasets/Sales.xlsx>

Sinh viên sử dụng Pandas và thực hiện các yêu cầu sau:

- (1) Xuất toàn bộ dữ liệu dữ liệu Sales ra màn hình console
- (2) Thông kê tuần có Sales_Volume cao nhất
- (3) Xuất các tuần có Sales_Volume từ X tới Y, với X, Y nhập từ bàn phím
- (4) Xuất tuần có Ads_Cost thấp nhất
- (5) Thiết kế giao diện và hiển thị dữ liệu bảng Sales lên QTableWidget

Hướng dẫn:

Sinh viên xem lại kiến thức được trình bày trong môn học và tìm hiểu thêm trên mạng để thực hiện bài làm.

Câu 126: Phân tích dữ liệu TCB - Pandas

Yêu cầu:

Cho Dataset: https://tranduythanh.com/datasets/TCB_2018_2020.csv

Sinh viên sử dụng Pandas và thực hiện các yêu cầu sau:

- (1) Xuất toàn bộ dữ liệu dữ liệu TCB ra màn hình console
- (2) Lọc ra các dữ liệu có Close <x> và Close <y>. Với x, y nhập từ bàn phím
- (3) Trích lọc dữ liệu chỉ lấy Date, High, Low và giá trị Low từ x tới y. Với x, y nhập từ bàn phím
- (4) Nhập vào 1 ngày, xuất thông tin chi tiết của ngày giao dịch này
- (5) Nhập vào 1 mảng các ngày, lọc dữ liệu theo mảng ngày này.

(6*) Thiết kế giao diện và hiển thị dữ liệu bảng TCB lên QTableWidget. Cung cấp các nút lệnh và giao diện khác để tương tác và hiển thị kết quả. Sinh viên tự đề xuất giao diện

Hướng dẫn:

Sinh viên xem lại kiến thức được trình bày trong môn học và tìm hiểu thêm trên mạng để thực hiện bài làm.

Câu 127: Phân tích giá trị - Pandas

Yêu cầu:

Cho Dataset: <https://tranduythanh.com/datasets/SampleData2.csv>

Sinh viên sử dụng Pandas và thực hiện các yêu cầu sau:

- (1) Xuất toàn bộ dữ liệu dữ liệu ra màn hình console
 - (2) Xuất toàn bộ dữ liệu nhưng sắp xếp theo Price tăng dần
 - (3) Nhập vào Symbol để tìm kiếm, nếu tìm thấy thì giảm Price xuống 1/2
 - (4) Viết mã lệnh bổ sung thêm cột USD, với giá trị = cột Price/23
 - (5) Viết mã lệnh cho người người sử dụng nhập Symbol, Price, PE, USD và nhập dữ liệu này vào cuối Data Frame
 - (6) Viết mã lệnh nhóm dữ liệu theo cột Group và thực hiện các lệnh thống kê cơ bản: mean, sum, count,.....
 - (7) Viết mã lệnh xóa dòng dữ liệu theo Symbol nhập từ bàn phím
- (8*) Thiết kế giao diện và hiển thị dữ liệu lên QTableWidget. Cung cấp các nút lệnh và giao diện khác để tương tác và hiển thị kết quả. Bổ sung các biểu đồ. Sinh viên tự đề xuất giao diện

Hướng dẫn:

Sinh viên xem lại kiến thức được trình bày trong môn học và tìm hiểu thêm trên mạng để thực hiện bài làm.

Câu 128: Xử lý dữ liệu rỗng - Pandas**Yêu cầu:**

Cho Dataset: https://tranduythanh.com/datasets/SampleData_NaN.csv

Sinh viên sử dụng Pandas và thực hiện các yêu cầu sau:

- (1) Xuất toàn bộ dữ liệu dữ liệu ra màn hình console
- (2) Xuất toàn bộ dữ liệu nhưng thay các giá trị RỖNG bằng TRUE
- (3) Viết lệnh kiểm tra số lượng dữ liệu rỗng cho toàn bộ DataFrame
- (4) Viết mã lệnh xóa các dòng chứa phần tử rỗng
- (5) Viết mã lệnh điền giá trị thay thế cho phần tử rỗng (nhập từ bàn phím)
- (6) Viết mã lệnh điền giá trị thay thế cho phần tử rỗng bằng các phương pháp **bfill**, **ffill** và **nội suy(interpolate)**
- (8*) Thiết kế giao diện và hiển thị dữ liệu lên QTableWidgetItem. Cung cấp các nút lệnh và giao diện khác để tương tác và hiển thị kết quả. Sinh viên tự đề xuất giao diện

Hướng dẫn:

Sinh viên xem lại kiến thức được trình bày trong môn học và tìm hiểu thêm trên mạng để thực hiện bài làm.

Câu 129: Trình bày cách cài đặt và sử dụng thư viện Matplotlib & Seaborn**Yêu cầu:**

Sinh viên hãy trình bày cách cài đặt và sử dụng thư viện Matplotlib & Seaborn. Cho một vài ví dụ liên quan tới xử lý trực quan hóa dữ liệu bằng các thư viện này, chẳng hạn như vẽ biểu đồ cột, biểu đồ pie...

Hướng dẫn:

Sinh viên xem lại kiến thức được trình bày trong môn học và tìm hiểu thêm trên mạng để thực hiện bài làm.

Câu 130: Trực quan hóa NetProfit với Matplotlib & Seaborn

Yêu cầu:

Cho Dataset: <https://tranduythanh.com/datasets/NetProfit.csv>

Sinh viên hãy thực hiện các lệnh trực quan hóa dữ liệu như trong bài học.

(1) Trực quan hóa single line: df[['Year', 'VIC']]



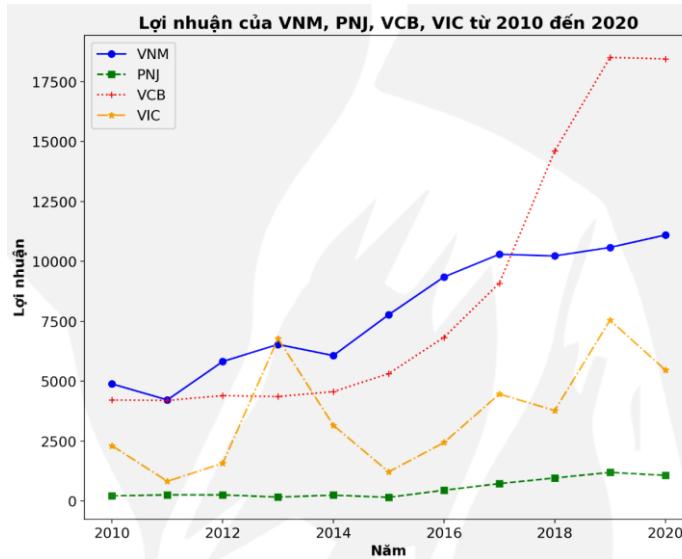
(2) Trực quan hóa multiple lines:

['Year', 'VNM']

['Year', 'PNJ']

['Year', 'VCB']

['Year', 'VIC']



Hướng dẫn:

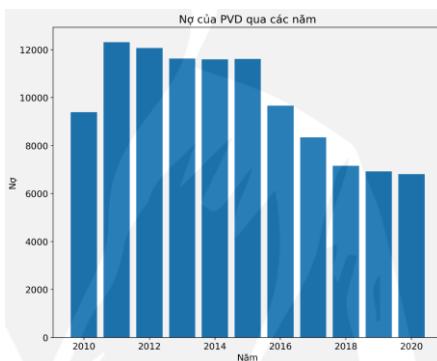
Sinh viên xem lại kiến thức được trình bày trong môn học và tìm hiểu thêm trên mạng để thực hiện bài làm.

Câu 131: Trực quan hóa PVD Assets với Matplotlib & Seaborn**Yêu cầu:**

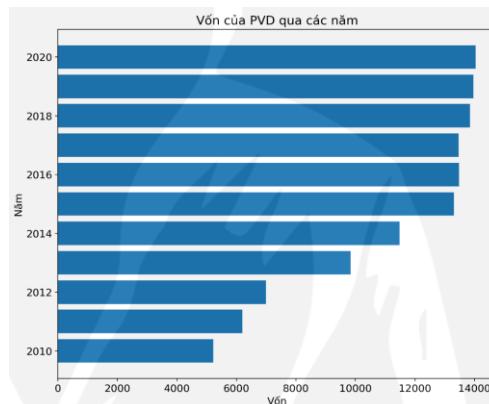
Cho Dataset: https://tranduythanh.com/datasets/PVD_Asset.csv

Sinh viên hãy thực hiện các lệnh trực quan hóa dữ liệu như trong bài học.

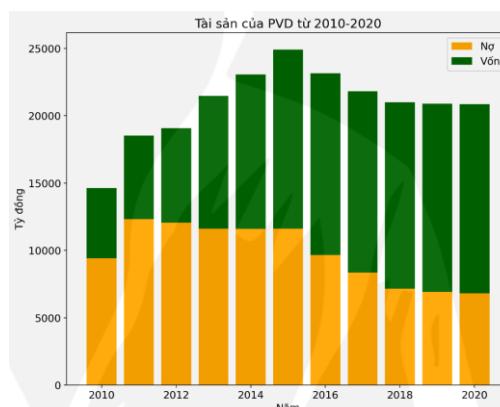
(1) Trực quan hóa Nợ của PVD qua các năm (Year, Liabilities)



(2) Trực quan hóa Vốn của PVD qua các năm (Year, Equity):



(3) Trực quan hóa Tài sản của PVD từ 2010-2020 (Year, Liabilities, Equity):



Hướng dẫn:

Sinh viên xem lại kiến thức được trình bày trong môn học và tìm hiểu thêm trên mạng để thực hiện bài làm.

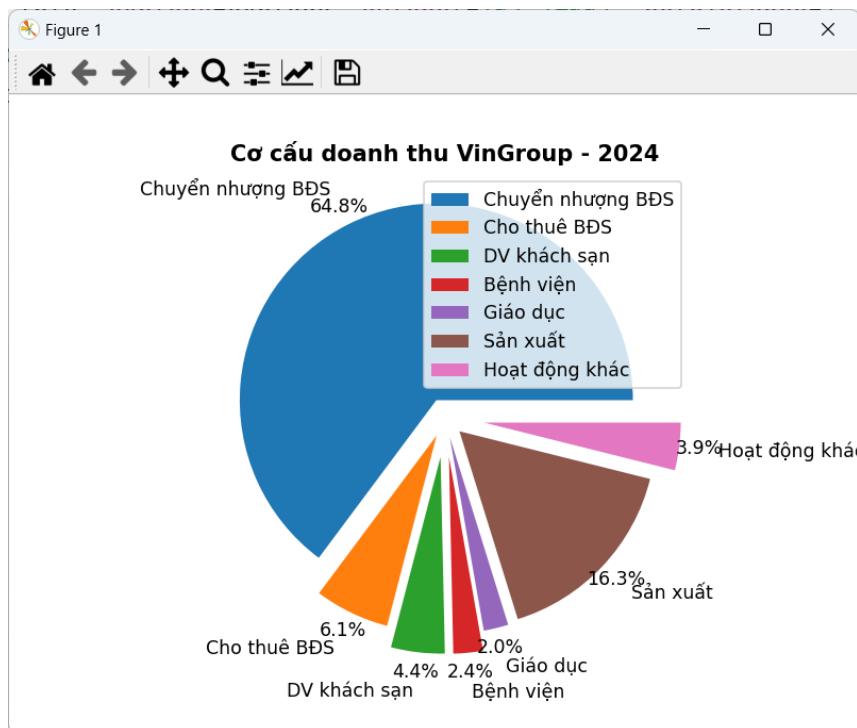
Câu 132: Trực quan hóa cơ cấu doanh thu với Matplotlib & Seaborn vào PyQt6

Yêu cầu:

Vingroup cho tập dữ liệu và nhãn sau:

```
1 | lbls = ['Chuyển nhượng BDS', 'Cho thuê BDS', 'DV khách sạn', 'Bệnh  
2 | viện', 'Giáo dục', 'Sản xuất', 'Hoạt động khác']  
3 | income = [71.576, 6.788, 4.869, 2.675, 2.244, 18.007, 4.304]  
4 | explode = [0.1, 0.1, 0.2, 0.2, 0.1, 0.1, 0.2]
```

Hãy vẽ biểu đồ hình tròn để trực quan hóa cơ cấu doanh thu của Vingroup:



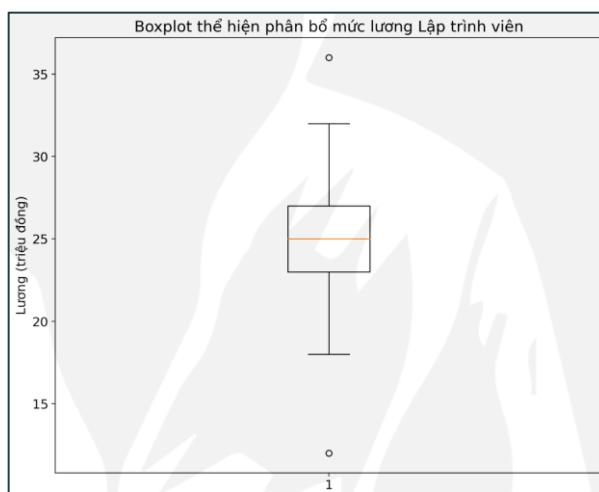
Hướng dẫn:

Sinh viên xem lại kiến thức được trình bày trong môn học và tìm hiểu thêm trên mạng để thực hiện bài làm.

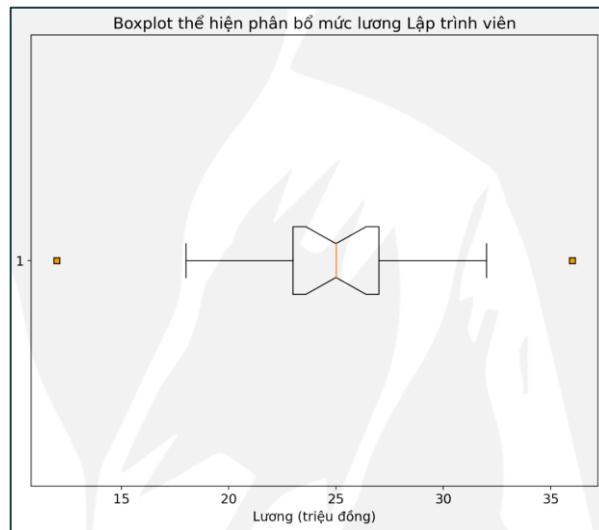
Câu 133: Trực quan hóa mức lương với Matplotlib & Seaborn vào PyQt6**Yêu cầu:**

Cho Dataset: https://tranduythanh.com/datasets/Salary_of_Developer.csv

- (0) Nạp toàn bộ dữ liệu và xuất ra màn hình
- (1) Trực quan hóa phân bố mức lương của lập trình viên bằng Boxplot nằm đứng



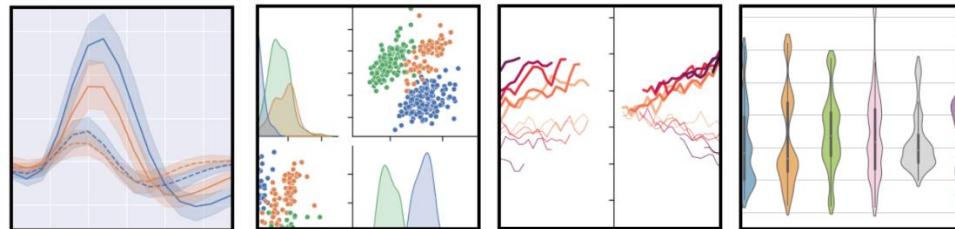
- (2) Trực quan hóa phân bố mức lương của lập trình viên bằng Boxplot nằm ngang

**Hướng dẫn:**

Sinh viên xem lại kiến thức được trình bày trong môn học và tìm hiểu thêm trên mạng để thực hiện bài làm.

Câu 134: Re-practice Matplotlib & Seaborn vào PyQt6**Yêu cầu:**

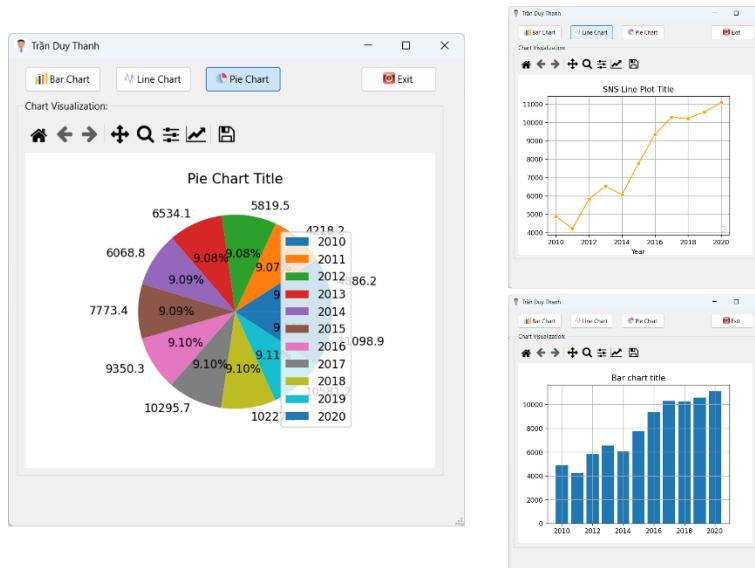
Sinh viên thực hiện lại tất cả các bài minh họa mẫu về SeaBorn trong slide bài học.

**Hướng dẫn:**

Sinh viên xem lại kiến thức được trình bày trong môn học và tìm hiểu thêm trên mạng để thực hiện bài làm. Xem từ Slide 103 tới slide 122 phần Thư viện trong Python.

Câu 135: Trình bày cách tích hợp Matplotlib & Seaborn vào PyQt6**Yêu cầu:**

Sinh viên hãy trình bày cách tích hợp Matplotlib & Seaborn vào PyQt6. Cho ví dụ minh họa, chẳng hạn như cách tích hợp Bar Char, Line Chart, Pie Chart vào PyQt6



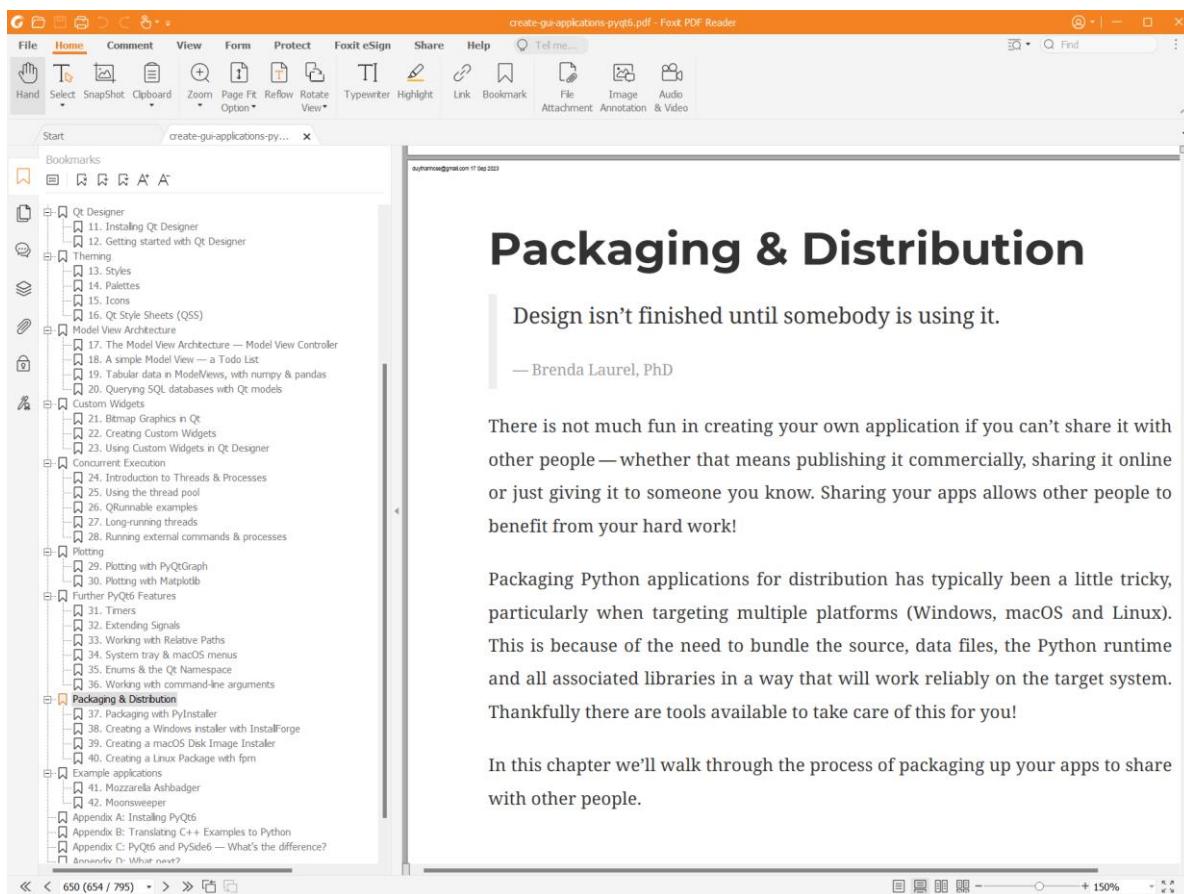
Hướng dẫn:

Sinh viên xem lại kiến thức được trình bày trong môn học và tìm hiểu thêm trên mạng để thực hiện bài làm. Xem từ slide 123 tới slide 130 phần thư viện trong Python.

Câu 136: Trình bày cách đóng gói và xuất bản phần mềm

Yêu cầu:

Sinh viên hãy trình bày cách đóng gói và xuất bản phần mềm viết bằng Python.



Martin Fitzpatrick (2022), Create GUI Applications with Python & Qt6, Link download: <https://tranduythanh.com/ebooks/create-gui-applications-pyqt6.pdf>

Hướng dẫn:

Sinh viên xem lại kiến thức được trình bày trong môn học và tìm hiểu thêm trên mạng để thực hiện bài làm.