## Code For Teens - Mini Hackathon 2

**Lưu ý**: phát bài tập cho học viên theo từng part, không đưa cho học viên link cuả toàn bộ bài tập.

**Mini hack list**

PART 1

1. Init color list - Viết chương trình chứa 1 list chứa ít nhất 3 string đại diện cho 3 màu, ví dụ: “blue”, “red", “teal", “green”
2. Print color list - Sử dụng lại list ở bài số 1, viết chương trình hiện ra tất cả các màu trong list

Ví dụ:

Our color list:

blue, red, teal, green

1. Create - Sử dụng lại list ở bài 1, viết chương trình hỏi người dùng muốn thêm màu gì vào list, sau khi người dùng nhập, thêm màu này vào cuối list

Ví dụ:

Our color list:

blue, red, teal, green

Enter a new color: **orange**

Our new color list:

blue, red, teal, green, **orange**

PART 2

1. Create (2) - Sử dụng lại list ở bài số 1, cho phép người dùng xem nội dung của 1 màu theo vị trí:

Ví dụ:

Enter position: **2**

Color at position 2: **red**

1. Delete - Sử dụng list ở bài 1, viết chương trình hỏi người dùng muốn xoá màu nào trong danh sách, nếu người dùng nhập số, tự động thực hiện xoá theo vị trí, nếu người dùng nhập chữ, tự động thực hiện xoá theo nội dung

Ví dụ:

Lần chạy 1:

Our color list:

1.blue

2.red

3.teal

4.green

Item to delete: **2**

Our new color list:

1.blue

2.teal

3.green

Lần chạy 2:

Our color list:

1.blue

2.red

3.teal

4.green

Item to delete: **teal**

Our new color list:

1.blue

2.red

3.green

1. Dash - Sử dụng turtle để vẽ một đường thẳng bao gồm các đoạn nhỏ nối liền nhau, mỗi đoạn có màu lần lượt là các màu trong danh sách ở bài 1. [Setup code](https://gist.githubusercontent.com/qhuydtvt/1deb89cd14c35cacc50fc5f82eb03852/raw/983cc781626c0d377501a7d3b833c4ba97e94b0b/turtle_setup.py)

PART 3

1. Search number in list - Tạo 1 list chứa trên 5 số nguyên không cách đều, sau đó yêu cầu nhập vào một số, thực hiện tìm kiếm số mà người dùng nhập vào trong list vừa khởi tạo, trả lời câu hỏi: “Số này có trong list không, nếu có thì có đứng thứ mấy trong dãy?”

Ví dụ: Giả sử list cho sẵn là 5, 1, 8, 92, -1, 30

Lần chạy 1:

Enter a number: -99

Not found in our list

Lần chạy 2:

Enter a number: 8

Found, position 8

1. Sum number in list - Tạo 1 list chứa trên 5 số không cách đều. Tính tổng dãy này và in ra kết quả. Thực hiện bài này bằng 2 cách, dùng hàm sum() và không dùng hàm sum() .

Ví dụ: Giả sử list cho sẵn là 5, 1, 8, 92, -1, 30

Sum of all numbers: 135

1. Sum number in list (2) - Thực hiện lại bài 8 với danh sách các số nhập từ người dùng, các số cách nhau bởi dấu cách ‘ ‘

Ví dụ:

Enter a list of numbers, separated by ‘ ‘: 5 12 6 61 124

Sum of all entered numbers: 208

PART 4

1. Filter even - Tạo 1 list chứa trên 5 số không cách đều. Lọc ra tất cả những số chẵn trong dãy này và in ra màn hình.

Ví dụ: Giả sử list cho sẵn là 5, 1, 8, 92, 7, 30

All even numbers: 8, 92, 30

1. Sequence input - Thực hiện lại bài 10 với danh sách các số nhập từ người dùng, các số cách nhau bởi dấu phẩy ‘,’

Ví dụ:

Enter a list of numbers, separated by ‘,’: 5, 12, 6, 61, 124

All even numbers from entered list: 12, 6, 124

PART 5

1. Init list of district names and population - Cho danh sách các quận của một thành phố cùng diện tích cũng như dân số của các quận này như bảng

Tạo 2 list:

* List đầu tiên, theo thứ tự từ trên xuống dưới, bao gồm tên của các quận trong bảng
* List thứ hai, theo thứ tự từ trên xuống dưới, bao gồm dân số của các quận trong bảng

|  |  |  |
| --- | --- | --- |
| **Quận** | **Km2** | **Dân số** |
| ST | 117,43 | **150,300** |
| BĐ | 9.224 | **247,100** |
| BTL | 43.35 | **333,300** |
| CG | 12.04 | **266,800** |
| ĐĐ | 9.96 | **420,900** |
| HBT | 10.09 | **318,000** |

1. Search max n min population - Ở trong list dân số của các quận trong bảng, tìm ra **chỉ số** của quận có số dân lớn nhất và quận có dân số ít nhất
2. From district index to name - Từ chỉ số của quận có dân số đông nhất và ít nhất, in ra tên của quận có số dân lớn nhất và số dân ít nhất

PART 6

1. Population density - Ở bảng bài số 12, tạo ra 1 list chứa diện tích của từng quận, theo thứ tự từ trên xuống dưới

Từ list chứa diện tích và list chứa dân số của các quận, tạo ra 1 list mới chứa mật độ dân cư

của các quận từ trên xuống. Công thức: Mật độ dân cư = Dân số / diện tích

1. Average population density - Từ list ở bài 15, tính mật độ dân cư trung bình của các quận. Công thức: Mật độ dân cư trung bình = Tổng mật độ dân cư / tổng số quận

PART 7

1. Init high score list - Tạo 1 list, chứa các số nguyên, đại diện cho điểm cao (High scores) của người chơi

Ví dụ: 45, 67, 56, 78

1. Print high score list - Hiện ra danh sách điểm cao (chưa cần phải đúng thứ tự từ trên xuống dưới ngay):

Ví dụ:

High scores:

1. 78

2. 56

3. 67

4. 45

1. New high score - Viết chương trình thêm điểm mới của người chơi (chưa cần phải đúng thứ tự từ trên xuống dưới ngay)

Ví dụ:

High scores:

1. 78

2. 56

3. 67

4. 45

Enter your new score: 70

High scores:

1. 78

2. 56

3. 67

4. 45

5. 70

PART 8

1. Sort high score - Thực hiện lại bài số 18 và 19 với các điểm được hiện từ cao đến thấp

Ví dụ:

Hiển thị điểm:

High scores:

1. 78

2. 67

3. 56

4. 45

Thêm điểm

High scores:

1. 78

2. 67

3. 56

4. 45

Enter your new score: 70

High scores:

1. 78

2. 70

3. 67

4. 56

5. 45

1. Sort and select 5 highest scores - Thực hiện lại bài 19 với các điểm từ cao đến thấp VÀ chỉ hiện ra điểm của 5 người điểm cao nhất

Ví dụ:

Lần chạy 1:

High scores:

1. 78

2. 67

3. 56

4. 45

Enter your new score: 70

High scores:

1. 78

2. **70**

3. 67

4. 56

5. 45

Lần chạy 2:

High scores:

1. 78

2. 70

3. 67

4. 56

5. 45

Enter your new score: 76

NEW High scores:

1. 78

2. **76**

3. 70

4. 67

5. 56