farm_table

January 23, 2023

1 1

Store a list of fruits and vegetables available from the farm in separate arrays (e.g. fruits = [“apple”, “orange”, “banana”] and vegetables = [“tomatoes”, “carrots”, “kale”]).

[ ]:
```
fruits = ["apple", "orange", "banana", "cherry", "pineapple"]
vegetables = ["tomatoes", "carrots", "kale", "potato", "broccoli"]
```

2 2

Use indexing and slicing to access specific elements in the arrays and print them out.

[ ]:
```
print("List of avaible fruits:")
for fruit in fruits:
    print (fruit)


print("\nList of avaible vegetables:")
for veggy in vegetables:
    print (veggy)
```

3 3

Create a nested list that stores different dishes that can be made using the fruits and vegetables (e.g. [[“tomato soup”, “tomatoes”], [“carrot juice”, “carrots”]]).

[ ]:
```
veggies_dishes = {"tomatoes soup": "tomatoes", "fries": "potato", "broccoli␣
 ↪soup": "broccoli", "steamed broccoli" : "broccoli", "salad" : ["kale",␣
 ↪"tomatoes"]}
fruits_dishes = {"Unknown": [], "apple juice": "apple", "orange juice":␣
 ↪"orange", "apple pie": "apple", "banana smoothie": ["banana", "orange"]}
```

4 4

Use iterable unpacking to print out the name of the dish and the ingredients it requires.

[ ]:
```
for i in veggies_dishes:
    print("The name of the dishe is: " + i)
    print("The ingredients are: " + str(veggies_dishes[i]))

for k in fruits_dishes:
    print("The name of the dishe is: " + k)
    print("The ingredients are: " + str(fruits_dishes[k]))
```

5 5

Use the enumerate() function to print out the index and the name of each dish.

[ ]:
```
for i, j in enumerate(veggies_dishes):
    print("The index of the dish is: " + str(i))
    print("The name of the dish is: " +  j)
print("\n")
for i, j in enumerate(fruits_dishes):
    print("The index of the dish is: " + str(i))
    print("The name of the dish is: " +  j)
```

6 6

Use the range() function to loop through the nested list and print out only the dishes that use a certain ingredient (e.g. only dishes that use tomatoes).

[ ]:
```
for i in veggies_dishes:

    if "tomatoes" in veggies_dishes[i]:
        print(i)
```

7 7

Use a list comprehension to create a new list of dishes that use at least one fruit.

[ ]:
```
new_fruits_dishes = [x for x in fruits_dishes if len(fruits_dishes[x]) > 0]
```

[ ]:
```
new_fruits_dishes
```

8 8

Create a generator expression that generates the name of the dish and the ingredients it requires for all dishes that use a certain ingredient (e.g. all dishes that use kale)

[ ]:
```
def generate(ingredient, dishes):
    for i in dishes:
        if ingredient in dishes[i]:
```

```
            print(f"The name of the dish that use {ingredient}: {i} "  )
            print(f"the ingredients: {dishes[i]}")
```

[ ]:
```
generate("banana", fruits_dishes)
```

9 Prime number:

A prime number is divisible by two factors (1 and itself). For example, 7 is a prime number because it is only divisible by 1 and 7. If the user enters an integer that's not between 1 and 5000, the program should display an error message. If the number is a prime number, the program should display a message to that effect. If the number is not a prime number, the program should display a message to that effect. Then, it should display the number of factors for the number and a list of those factors. Store the factors for each number in a list. Use functions to organize the code for this program.

[2]:
```
check = True

while check:
    number = int(input("Enter a nunber between 1 and 5000: "))
    num_factors = []
    if 0 < number <= 5000:
        for i in range(1, 5001):
            if number % i == 0:
                num_factors.append(i)
        if len(num_factors) == 2:
            print(f"{number} is a prime number")
            go = input("Try again Y/N: " )
            if go.lower() ==  "n":
                print("By!")
                break
        else:
            print(f"{number} is NOT a prime number")
            print(f"It has {len(num_factors)} factors: {num_factors}" )
            go = input("Try again Y/N: " )
            if go.lower() ==  "n":
                print("By!")
                break
    else:
        print("Error")
        number = int(input("Enter a nunber between 1 and 5000: "))
```

```
Enter a nunber between 1 and 5000:  5

5 is a prime number

Try again Y/N:  y
Enter a nunber between 1 and 5000:  45
```

```
45 is NOT a prime number
It has 6 factors: [1, 3, 5, 9, 15, 45]

Try again Y/N:  n

By!
```

[ ]: