

(‘Bake It Till You Make It’)



[A Python Program]



Created By: Tianna Fougeray



Summer 2022 Python Final Project

Table Of ‘Contents’

01 Program Background

The recipe for success

02 How It Works

The ingredients



03 Final Results

The final product

A black wire cooling rack holds five round, golden-brown donuts with a light dusting of sugar. In the bottom right corner, a silver metal bowl contains a single similar donut, with a hand featuring red-painted fingernails holding it from below.

01

Program Background

Build a Python program that combines desserts and workouts? Piece of cake!

Are your dessert cravings at an all-time high? Are you fresh out of unique combinations to bake? Oh, and did you make sure to get your daily workout in today? With ‘Bake It Till You Make It,’ you *can* have your cake and eat it too...

Welcome to ‘Bake It Till You Make It’ - a Python program designed for the dessert-and-fitness-obsessed grad students who are looking for new and delicious sweet treats to fuel their busy schedules and workouts.

This program utilizes pandas standard library for data manipulation, analysis, and easy reading from csv files. It also has its own custom library (called `myfinallibrary.py`) with functions, classes, and constants. Imports were made from the custom library to my standard Pandas library. The main program that you will run is titled `mainprog.py`.



1

The Data

2

3

Dessert Flavor Combinations

4

- <https://www.kaggle.com/datasets/keytarrockstar/dessert-flavor-combinations>
- 7,486 unique dessert combinations by flavor and rating

5

Workout Exercises

6

- <https://airtable.com/shrKZ9lPpw7EvjZ3X/tblvscpkbagqlWkkH>
- 88 unique exercises by equipment, exercise type, and muscle group

02

How It Works



Custom Library

myfinallibrary.py

```
1  
2  
3  
4          Function for Welcome statements:  
5 # custom library with welcome function, classes used for the exercise data, and constant  
6 # creating a welcome function  
7 def welcome():  
8     print('Welcome to Bake it Till You Make It! A sweet treat awaits you...')  
9     print('First, here is a little preview of 4-star rating cakes and cupcakes!')  
10  
11  
12  
13  
14
```

Custom Library

myfinallibrary.py

```
1 # class for Equipment and object orientation
2
3 class Equipment:
4
5     def __init__(self, equipment_name, equipment_quantity):
6         self.equipment_name = equipment_name
7         self.equipment_quantity = equipment_quantity
8
9     # class for Dumbbell/inheritance - will show quantity of equipment and weight
10    class Dumbbell(Equipment):
11
12        def __init__(self, equipment_name, equipment_quantity, weight):
13            super().__init__(equipment_name, equipment_quantity)
14            self.weight = weight
15
16    # class for Kettlebell/inheritance - will show quantity of equipment and weight
17    class Kettlebell(Equipment):
18
19        def __init__(self, equipment_name, equipment_quantity, weight):
20            super().__init__(equipment_name, equipment_quantity)
21            self.weight = weight
22
23    # class for Barbell/inheritance - will show quantity of equipment and weight
24    class Barbell(Equipment):
25
26        def __init__(self, equipment_name, equipment_quantity, weight):
27            super().__init__(equipment_name, equipment_quantity)
28            self.weight = weight
```

```
# sets quantity and weight in pounds for all equipment
dumbbell = Dumbbell("Dumbbell", 2, 15)
kettlebell = Kettlebell("Kettlebell", 1, 25)
barbell = Barbell("Barbell", 1, 40)

# constant
Constant = "Enjoy your sweet treat and workout!"
print(Constant)
```

Object orientation programming
-classes + inheritance
(explained in more depth on slide 20)

Save Room for Dessert

1
2 Lines of Python code and
3 baking a dessert have a lot
4 in common. All it takes is
5 knowing where to begin and
6 end with a step-by-step
7 process (your recipe),
8 adding in the proper code
9 to make the process work
(your ingredients), and
finally putting it all
together (your incredible
freshly baked dessert!)

10
11 Pictured to the right are
12 the columns used in our
13 dessert flavor combinations
14 dataset

recipe_name	flavors	rating
Big Apple Crumble Cupcakes	cinnamon cinnamon apple cinnamon vanilla	0
Bacon-Latticed Apple Pie	apple lemon cinnamon nutmeg	0
Lemon and Fig Cupcakes	vanilla lemon lemon fig vanilla	0
Toasted Coconut Chia Pudding	almond coconut	0
Coconut-Strawberry Ice Cream Pie	coconut graham coconut coconut strawberry strawberry...	4
Apple Walnut Bundt Cake	walnut cinnamon apple vanilla caramel	4
Caramel Sauce and Glaze	vanilla	0
Sweet Potato Sonker with Milk Dip	apple vanilla	0
Peach Lassi Sorbet with Crushed Blackberries	peach vodka honey blackberry lemon	0
Plum-Cardamom Crumble with Pistachios	plum lemon lemon cinnamon pistachio	0
Ricotta Panna Cotta with Nectarines and Honey	ricotta vanilla honey	0
Matcha-Chocolate Chip Cookies	coconut coconut vanilla banana oat chocolate matcha	0
Matcha Coconut Ice Cream	matcha coconut maple vodka vanilla coconut	0
Date Ice Cream (Buza 'Ala-Tamr)	rose cardamom date pistachio	2
Lemon and Lime Battenberg	almond lemon lime lime	0
Out-of-This-World Cherry Pie	cherry lemon	0
Cinnamon-Oat Peach Crisp	peach oat cinnamon vanilla	4
Mini Strawberry Eton Mess	strawberry	0
Blackberry Icebox Cake	blackberry blackberry lemon vanilla	0
Matcha-Dipped Strawberries	chocolate coconut matcha strawberry	0
Big-Batch Strawberry Compote	strawberry lemon lemon	4
Southern "One-Cup" Peach Cobbler	peach lemon lemon vanilla vanilla	3.5
Rhubarb Custard Cake	rum lemon rhubarb	3.5
Brioche au Rhum	rum orange vanilla fig	0
Italian Sundaes with Nutella	hazelnut	0

Let's Break It Down...

- 1
- 2
- 3 01 Program opens with a list of the highest 4-star ratings cupcakes and cakes (pictured to the right)
- 4
- 5 02 The user is prompted to type a dessert flavor
- 6
- 7
- 8 03 The user is asked if they would like to type an additional dessert flavor or they can type 'N' for no
- 9
- 10
- 11 04 A list of desserts from the `dessertdataset.csv` file is returned that match those flavor keywords chosen
- 12
- 13 05 An output file with the dessert recipes titled `dessert_output.csv` is printed to the user's computer
- 14

Enjoy your sweet treat and workout!	
Welcome to Bake it Till You Make It! A sweet treat awaits you...	
First, here is a little preview of 4-star rating cakes and cupcakes!	
None	
Top rated cakes and cupcakes:	
Recipe	Rating
Apple Walnut Bundt Cake	4.0
Ricotta Cheesecake with Almond Praline	4.0
Olive Oil Cake	4.0
Orange Cranberry Cupcakes	4.0
Toasted Marshmallow Cupcakes	4.0
Classic Carrot-Coconut Cake	4.0
Hungarian Golden Pull-Apart Cake with Walnuts and Apricot Jam (Aranygaluska)	4.0
Coco Lopez Coconut Cake	4.0
Dark Ginger Rye Cake With Yogurt and Honey	4.0
Fancy Dirt Cake	4.0
Lemon Blossom Cupcakes	4.0
Lemon Blueberry Cupcakes with a Citrus Glaze	4.0
Blueberry-Lemon Icebox Cake	4.0
Vanilla-Buttermilk Wedding Cake With Raspberries and Orange Cream-Cheese Frosting	4.0
BA's Best Carrot Cake	4.0
Cherry-Cornmeal Upside-Down Cake	4.0
Grapefruit-Poppy Seed Loaf Cake with Yogurt Glaze	4.0
Apple Jack Stack Cake	4.0
Classic Lemon Cheesecake	4.0
Food Processor Carrot Cake	4.0

The Code Behind It - mainprog.py

1 Imports for pandas and custom library

2

3 Outfile creates an outfile for writing results that will be saved as `dessert_output.csv`, and later on a `results.txt` file at the end of the program (File I/O)

4

5 Pd.read ensures csv file is read properly

6

7 Print statement prints the top cakes and cupcakes

8

9

10

11

12

13

14

```
# import from standard pandas library
import pandas as pd
from myfinallibrary import welcome

# create file for writing results
out_file = open("results.txt", "w", encoding='utf-8')

# reads the dessert csv file
df1 = pd.read_csv('dessertdataset.csv')
print(welcome())

# calculate and display top cakes/cupcakes before prompting flavors with rating 4
top_cakes = df1[df1['rating'] == 4]
top_cakes = top_cakes[top_cakes['recipe_name'].str.contains('Cake|cake|CAKE')]
print('Top rated cakes and cupcakes:')
print("{:<10} {:<9}".format("Recipe", "Rating"))
for ind in top_cakes.index:
    print("{:<10} {:<9}".format(top_cakes['recipe_name'][ind], top_cakes['rating'][ind]))
print('*****')
```

The Code Behind It - mainprog.py

- 1 Flavors is defined at the very beginning
- 2
- 3 Iteration and decision making is used to search through the dessert flavors depending on what flavor(s) the user gives it
- 4
- 5
- 6 The loop will stop once the user types 'N' when prompted
- 7
- 8 Outfile.write ensures the dessert header will write to the outfile
- 9
- 10
- 11
- 12
- 13
- 14

```
# define flavors
flavors = []
print('Time to find your next favorite dessert...')
# iteration
while True:
    # user types one flavor
    flavors.append(input('Type a dessert flavor: ').lower())
    # user types additional flavors if they choose. loop will stop once they type N
    add_another_flavor = input('Do you want to add another flavor? Type the flavor or N: ')
    if add_another_flavor.lower() != 'y':
        break
# write dessert header to output file
out_file.write("Desserts:\n")
```

The Code Behind It - mainprog.py

1 Print statement prints the
2 dessert recipes by using a
3 **Data Structure** list and also
4 informs the user that they
will select workouts in the
next part of the program

5 Pandas dataframe is used to
6 create outfile of the
7 dessert recipes in a single
list

8 Error handling was tested in
9 case the user received an
error to inform the user of
what caused the error

```
# prints all dessert combos in data set with those flavor names
print('The dessert combinations you can make are...')
print('Scroll down...there are still workouts to pick!')

# data structure list
df_list = []
for ind in df1.index:
    select_dessert = True
    for selected_flavor in flavors:
        if selected_flavor not in str(df1['flavors'][ind]):
            select_dessert = False
    if select_dessert:
        # write the dessert to the out file
        df_list.append(df1['recipe_name'][ind])

    out_file.write(df1['recipe_name'][ind] + "\n")
# was researching and came across error handling... wanted to give it a try here
try:
    # Pandas dataframe that creates it off a single list of recipe names and outputs to csv file.
    df_test = pd.DataFrame(df_list, columns=['recipe_name'])
    print(df_test.reset_index(drop=True))
    df_test.to_csv("dessert_output.csv", index=False)
except:
    print("The dataframe could not be written, unknown error!")
```

Let's Try an Example!

- 1 1. Run mainprog.py
- 2 2. Type a dessert flavor:
 strawberry
- 3 Then, press ENTER
- 4 3. Do you want to add another
 flavor? Type the flavor or N:
 chocolate
- 5 Then, press ENTER again
- 6 4. Desserts that are
 strawberry and/or chocolate
 flavored will print
- 7 5. You will now have a file
 printed of the desserts and
 will move along to the second
 part of the program: selecting
 workouts
- 8
- 9
- 10
- 11
- 12
- 13
- 14

```
Time to find your next favorite dessert...
Type a dessert flavor: strawberry
Do you want to add another flavor? Type the flavor or N: chocolate
The dessert combinations you can make are...
Scroll down...there are still workouts to pick!
                                         recipe_name
0          Coconut-Strawberry Ice Cream Pie
1          Mini Strawberry Eton Mess
2          Matcha-Dipped Strawberries
3          Big-Batch Strawberry Compote
4          Strawberry-Rhubarb Galette with Buckwheat Crust
...
498        Hazelnut, Chocolate and Strawberry Torte
499        Pecan Torte with Strawberries and Cream
500        Strawberry Compote with Blood Orange Juice
501        Sam's Jell-O
502        Vanilla Ice Cream and Ginger Molasses Cookie S...
...
[503 rows x 1 columns]
*****
Your dessert results were printed as a separate file on your computer
Before you take a look at those, select a workout!
```

Eat. Sleep. Gym. Repeat.



Just like a workout,
you must consistently
put in the work if
you want to see
excellent results in
a Python program.

Pictured to the right
are the columns used
in our exercise
dataset

C1	C2	C3	C4	C5
Exercise	Equipment	Exercise Type	Major Muscle	Minor Muscle
Arnold Press	Dumbbells	Weight	Arms	Bicep,Shoulders
Bicep Curl	Bar,Dumbbells,Cable	Weight,Machine	Arms	Bicep
Bicycle Crunch	Body Weight	Weight	Core	<null>
Bounds	Body Weight	Cardio,Laps	Full Body,Legs	Outer Thigh,Glutes
Box Jumps	Body Weight	Plyo	Full Body	<null>
Box Toe Touch	Platform	Cardio	Legs	Glutes
Broad Jump	Body Weight	Plyo,Laps,Cardio	Legs	Glutes
Bulgarian Split Squat	Bar,Dumbbells	Weight	Legs	Hamstrings,Quads
Burpee	Body Weight,Bosu Ball	Plyo	Full Body	<null>
Burpee Broad Jump	Body Weight	Plyo,Laps	Full Body	<null>
Butt Kickers	Body Weight	Cardio	Legs	Glutes
Calf Raise	Dumbbells,Machine	Weight	Legs	Calves
Chest Press	Dumbbells	Weight	Arms	Chest
Close to Wide Grip Burnout	Dumbbells	Weight	Arms	Bicep
Compass Jump	Body Weight	Cardio	Full Body	<null>
Crab Crawl	Body Weight	Cardio,Laps	Full Body	<null>
Curtsey Lunges	Dumbbells,Bar	Weight	Legs	Inner Thigh
Deficit Squat	Platform	Weight	Legs	Glutes
Donkey Kick	Dumbbells,Band,Body Weight	Weight	Legs	Glutes
Fire Hydrant	Band,Body Weight	Weight	Legs	Glutes,Outer Thigh
Flutter Kick	Body Weight	Weight	Core	<null>
Frogger	Body Weight	Plyo	Full Body	<null>
Glute Bridge	Band,Dumbbells	Weight	Legs	Glutes
Glute Bridge March	Band,Platform	Weight	Legs	Hamstrings

Let's Break It Down...

- 1
- 2
- 3 01 Following the list of desserts printed, the program will ask the user
4 to type a major muscle group (choices given based on dataset)
- 5
- 6 02 The user will then be asked to type a minor muscle group (choices
7 given based on dataset)
- 8
- 9 03 The user is finally asked if they want to add another muscle group from
10 those choices above
- 11
- 12 04 A list of workouts from the `exercisedata.csv` file is returned
13 that match those muscle group keywords chosen as well as the
14 equipment to use with quantity needed and weight suggestions

The Code Behind It - mainprog.py

- 1 Major muscle groups and minor muscle groups are defined before the code is called
 - 2
 - 3
 - 4
 - 5 Pd.read ensures the workout list from the exercises they chose will appear in the results.txt file at the end
 - 6
 - 7
 - 8 Like the dessert portion of the program, **iteration** and **data structure lists** are used here to search through exercises depending on what major and minor muscle group(s) the user gives
 - 9
 - 10
 - 11
 - 12
 - 13 The loop will stop when the user types 'N'
 - 14
- ```
reading the exercise csv
df2 = pd.read_csv('exercisedata.csv')

data structure lists
major_muscle = []
minor_muscle = []

iteration
while True:
 # user types one muscle group
 major_muscle.append(input('Type a major muscle group (arms,legs,core,full body,or back)').lower())
 minor_muscle.append(input('Type minor muscle group (glutes,inner thigh,' +
 'outer thigh,quads,hamstrings,shoulders,tricep,bicep,chest or obliques:').lower())

 # user types additional muscle groups if they choose. loop will stop once they type N
 add_another_major_muscle = (input('Do you want to add another major muscle group? Type the muscle or N: '))
 if add_another_major_muscle.lower() != 'y':
 break
 add_another_minor_muscle = (input('Do you want to add another minor muscle group? Type the muscle part or N: '))
 if add_another_minor_muscle.lower() != 'y':
 break

 # write workouts header to output file
 out_file.write("\nWorkouts:\n")
```

# The Code Behind It - mainprog.py

- 1      The exercises that the user  
2      can do will print after the  
3      iteration
- 4      The import from our custom  
5      library ensures classes,  
6      inheritance, and use of  
7      object orientation for  
8      equipment will print
- 9      File I/O is completed for  
10     the results.txt file to save  
11     to the user's computer
- 12
- 13
- 14

```
prints all exercises in data set with those muscle group names
print('The exercises you can do are...')

for ind in df2.index:
 select_exercise = True
 if str(df2['Major Muscle'][ind]).lower() not in major_muscle and str(df2['Minor Muscle'][ind]).lower() not in minor_muscle:
 select_exercise = False
 if select_exercise:
 print(df2['Exercise'][ind])
 # write the workout to the outfile
 out_file.write(df2['Exercise'][ind] + "\n")

importing from my library program
from myfinallibrary import *

prints the equipment with the quantity and weight in (lbs.) suggestion
print('The workout equipment you can use by quantity and weight suggestion:')
print(dumbbell.equipment_name)
print('quantity needed:' ,dumbbell.equipment_quantity)
print("weight suggestion:" ,dumbbell.weight)
print(kettlebell.equipment_name)
print('quantity needed:' ,kettlebell.equipment_quantity)
print(kettlebell.weight)
print(barbell.equipment_name)
print('quantity needed:' ,barbell.equipment_quantity)
print('weight suggestion:' ,barbell.weight)
print('*****')
```

## The Code Behind It - mainprog.py

```
1 print('Check your desktop or file folder- your dessert and workout results were now printed to an output file!')
2 # close file once program is finished
3 out_file.close()
4
5 # constant from custom library
6 from myfinallibrary import Constant
7 print((Constant))
8
8 The finalized outfile is printed
9 with a list of workouts as well as
10 the desserts from the beginning. A
11 constant is used here to print a
12 closing statement in mainprog.py:
13
14
```

Check your desktop or file folder- your dessert and workout results were now printed to an output file!  
Enjoy your sweet treat and workout!

# The Code Behind It

## myfinallibrary.py

1 As the previous slides  
2 explained, our **classes** for  
3 the exercise data are stored  
4 in the **custom library**  
5 created

6 **Object orientation** was used  
7 for Equipment and  
8 specifically, **inheritance**  
9 was used for the 3 types of  
equipment in the dataset:  
dumbbell, kettlebell, and  
barbell

10 Weight in pounds were  
11 randomly chosen as  
12 suggestions and quantity  
13 defines how much of the  
14 equipment you may need to  
complete the exercise

```
class for Equipment and object orientation
class Equipment:

 def __init__(self, equipment_name, equipment_quantity):
 self.equipment_name = equipment_name
 self.equipment_quantity = equipment_quantity

class for Dumbbell/inheritance - will show quantity of equipment and weight
class Dumbbell(Equipment):

 def __init__(self, equipment_name, equipment_quantity, weight):
 super().__init__(equipment_name, equipment_quantity)
 self.weight = weight

class for Kettlebell/inheritance - will show quantity of equipment and weight
class Kettlebell(Equipment):
 def __init__(self, equipment_name, equipment_quantity, weight):
 super().__init__(equipment_name, equipment_quantity)
 self.weight = weight

class for Barbell/inheritance - will show quantity of equipment and weight
class Barbell(Equipment):
 def __init__(self, equipment_name, equipment_quantity, weight):
 super().__init__(equipment_name, equipment_quantity)
 self.weight = weight

sets quantity and weight in pounds for all equipment
dumbbell = Dumbbell("Dumbbell", 2, 15)
kettlebell = Kettlebell("Kettlebell", 1, 25)
barbell = Barbell("Barbell", 1, 40)
```

# Let's Try an Example!

```
1 1. Type a major muscle group (arms, legs, core,
2 full body, or back): legs
3 Then, press ENTER
4
5 2. Type a minor muscle group (glutes, inner thigh,
6 outer thigh, quads, hamstrings, shoulders, tricep,
7 bicep, chest, or obliques: glutes
8 Then, press ENTER
9
10 3. Exercises you can do for legs and glutes will
11 print
12
13 4. Keep scrolling down through the list of
14 exercises and you'll see these lines printed from
the custom library:

The equipment you can use by quantity and
weight suggestion:
Dumbbell
quantity needed: 2
weight suggestion: 15
Kettlebell
quantity needed: 1
25
Barbell
quantity needed: 1
weight suggestion: 40
```

```
Your dessert results were printed as a separate file on your computer
Before you take a look at those, select a workout!
Type a major muscle group (arms,legs,core,full body,or back): legs
Type minor muscle group (glutes,inner thigh,outer thigh,quads,hamstrings,shoulders, tricep, bicep, chest, or obliques: glutes
Do you want to add another major muscle group? Type the muscle or N: n
The exercises you can do are...
Box Toe Touch
Broad Jump
Bulgarian Split Squat
Butt Kickers
Calf Raise
Curtsy Lunges
Deficit Squat
Donkey Kick
Fire Hydrant
Glute Bridge
Glute Bridge March
Goblet Squat
Jump Lunges
Lateral Band Walk
Side Lunge
Single Leg Hip Bridge
Single Leg Squat
Skaters
Squat
Squat Jump
Squat to Lateral Leg Lift
Standing Glute Kickbacks
Standing Leg Lift
Step Up Lunges
Step-Back Lunge
Sumo Squat
```

03

## Final Results



1  
2      dessert\_output.csv  
3  
4  
5      Click to open  
6      dessert\_output.csv to  
7      view the dessert recipes  
8      generated from the  
9      flavors you gave Python  
10  
11     Scroll down the list for  
12     many more delicious  
13     desserts!  
14



A screenshot of a CSV file titled "dessert\_output.csv" in a spreadsheet application. The file contains a single column of dessert names. A yellow hand cursor is pointing at the file icon in the title bar.

| recipe_name                                            |
|--------------------------------------------------------|
| Coconut-Strawberry Ice Cream Pie                       |
| Mini Strawberry Eton Mess                              |
| Matcha-Dipped Strawberries                             |
| Big-Batch Strawberry Compote                           |
| Strawberry-Rhubarb Galette with Buckwheat Crust        |
| Coconut Custard and Rhubarb Tart                       |
| Meringue Sundae with Peppery Berry Sauce               |
| Retro Strawberries-and-Cream Pretzel Tart              |
| Oven-Dried Strawberries                                |
| Frosé (Frozen Rosé) Ice Pops                           |
| Easter Egg Cake with Strawberry Frosting               |
| Giant Meringue (Pavlova Gigantata)                     |
| Babee's Strawberry Shortbread Valentine's Cookies      |
| Strawberry Shortcake Cupcakes                          |
| Chocolate-Covered Marshmallow Cookies                  |
| Almond and Jam Tart                                    |
| Neo-Neapolitan Bombe                                   |
| Mixed Berry Pie Bars                                   |
| Frozen Peaches With Strawberries and Mint              |
| Watermelon Jell-O Shots                                |
| BA's Best Strawberry Shortcake                         |
| Strawberry Shortcake with Thyme and Whipped Cream      |
| Roasted Strawberry Trifles with Lemon Cream            |
| Strawberries With Balsamic Toffee                      |
| Coeur à la Crème with Roasted Strawberry Sauce         |
| Mochi-Covered Strawberries                             |
| Strawberry Shortcake Ice Cream Bars                    |
| Strawberry-Cucumber Salad with Lemon Cream             |
| Strawberry-Almond Cornmeal Cake                        |
| Sweet and Sour Strawberry Semifreddo with Black Sesame |
| Strawberry and Cranberry Popsicles                     |
| Coconut Milk Custard with Strawberry-Rhubarb Compote   |

1  
2     results.txt  
3  
4  
5     Click to open  
6     results.txt stored  
7     on your computer as  
8     well if you prefer  
9     to see the list of  
10    desserts and the  
11    workouts in one  
12    single file  
13  
14



results.txt

Desserts:

- Coconut-Strawberry Ice Cream Pie
- Mini Strawberry Eton Mess
- Matcha-Dipped Strawberries
- Big-Batch Strawberry Compote
- Strawberry-Rhubarb Galette with Buckwheat Crust
- Coconut Custard and Rhubarb Tart
- Meringue Sundae with Peppery Berry Sauce
- Retro Strawberries-and-Cream Pretzel Tart
- Oven-Dried Strawberries
- Frosé (Frozen Rosé) Ice Pops
- Easter Egg Cake with Strawberry Frosting
- Giant Meringue (Pavlova Gigantata)
- Babée's Strawberry Shortbread Valentine's Cookies
- Strawberry Shortcake Cupcakes
- Chocolate-Covered Marshmallow Cookies
- Almond and Jam Tart
- Neo-Neapolitan Bombe
- Mixed Berry Pie Bars
- Frozen Peaches With Strawberries and Mint
- Watermelon Jell-O Shots
- BA's Best Strawberry Shortcake
- Strawberry Shortcake with Thyme and Whipped Cream
- Roasted Strawberry Trifles with Lemon Cream
- Strawberries With Balsamic Toffee
- Cœur à la Crème with Roasted Strawberry Sauce
- Mochi-Covered Strawberries
- Strawberry Shortcake Ice Cream Bars
- Strawberry-Cucumber Salad with Lemon Cream
- Strawberry-Almond Cornmeal Cake

Workouts:

- Box Toe Touch
- Broad Jump
- Bulgarian Split Squat
- Butt Kickers
- Calf Raise
- Curtsey Lunges
- Deficit Squat
- Donkey Kick
- Fire Hydrant
- Glute Bridge
- Glute Bridge March
- Goblet Squat
- Jump Lunges
- Lateral Band Walk
- Side Lunge
- Single Leg Hip Bridge
- Single Leg Squat
- Skaters
- Squat
- Squat Jump
- Squat to Lateral Leg Lift
- Standing Glute Kickbacks
- Standing Leg Lift
- Step Up Lunges
- Step-Back Lunge
- Sumo Squat



Enjoy your sweet treat  
and workout!