QUBKitchen – Testing

part02

Hannah Dennis | 40406567

Contents

[Ingredient Tests 1](#_Toc152596895)

[test01 1](#_Toc152596896)

[test02 1](#_Toc152596897)

[test03 1](#_Toc152596898)

[test04 1](#_Toc152596899)

[test05 2](#_Toc152596900)

[test06 2](#_Toc152596901)

[test07 2](#_Toc152596902)

[test08 3](#_Toc152596903)

[test09 3](#_Toc152596904)

[test10 3](#_Toc152596905)

[test11 4](#_Toc152596906)

[test12 4](#_Toc152596907)

# Ingredient Tests

## test01

--- Test 1: Ingredient Constructor when valid -----------

Test data: 'Carrots', FoodType.FRUITVEG, 20

Ingredient constructed

After

ID: 0

Name: Carrots

Food Group: Fruit & Veg

Calories: 20

--- End Test Case -------------------------------------

## test02

--- Test 2: Recipe Constructor when invalid ----------

Test data: '123', null, 3000

Ingredient constructed

Error: java.lang.Exception:

Data could not be written

Reason(s):

java.lang.Exception: Name must be letters or characters: /&,.'-

java.lang.Exception: Invalid Food Group

java.lang.Exception: Calories must be 0-999

--- End Test Case -------------------------------------

## test03

--- Test 3: setName() when valid ----------------------

Before

ID: 2

Name: test

Food Group: Undefined

Calories: 1

Test data: 'Updated Name'

After

ID: 2

Name: Updated Name

Food Group: Undefined

Calories: 1

--- End Test Case -------------------------------------

## test04

--- Test 4: setName() when invalid -------------------

Before

ID: 3

Name: test

Food Group: Undefined

Calories: 1

Test data: '123'

After

ID: 3

Name: test

Food Group: Undefined

Calories: 1

Error: java.lang.Exception: Name must be letters or characters: /&,.'-

--- End Test Case -------------------------------------

## test05

--- Test 5: setName() when null ----------------------

Before

ID: 4

Name: test

Food Group: Undefined

Calories: 1

Test data: ''

After

ID: 4

Name: test

Food Group: Undefined

Calories: 1

Error: java.lang.Exception: Name cannot be blank or null

--- End Test Case -------------------------------------

## test06

--- Test 6: setFoodType() when valid -----------------

Before

ID: 5

Name: test

Food Group: Undefined

Calories: 1

Test data: FoodType.DAIRY

After

ID: 5

Name: test

Food Group: Dairy

Calories: 1

--- End Test Case -------------------------------------

## test07

--- Test 7: setFoodType() when invalid ---------------

Before

ID: 6

Name: test

Food Group: Undefined

Calories: 1

Test data: null

After

ID: 6

Name: test

Food Group: Undefined

Calories: 1

Error: java.lang.Exception: Invalid Food Group

--- End Test Case -------------------------------------

## test08

--- Test 8: setCalories() when valid ------------------

Before

ID: 7

Name: test

Food Group: Undefined

Calories: 1

Test data: 55

After

ID: 7

Name: test

Food Group: Undefined

Calories: 55

--- End Test Case -------------------------------------

## test09

--- Test 9: setCalories() lowest possible -------------

Before

ID: 8

Name: test

Food Group: Undefined

Calories: 1

Test data: 0

After

ID: 8

Name: test

Food Group: Undefined

Calories: 0

--- End Test Case --------------------------------------

## test10

--- Test 10: setCalories() highest possible -------------

Before

ID: 9

Name: test

Food Group: Undefined

Calories: 1

Test data: 999

After

ID: 9

Name: test

Food Group: Undefined

Calories: 999

--- End Test Case --------------------------------------

## test11

--- Test 11: setCalories() when invalid (low) ----------

Before

ID: 10

Name: test

Food Group: Undefined

Calories: 1

Test data: -1

After

ID: 10

Name: test

Food Group: Undefined

Calories: 1

Error: java.lang.Exception: Calories must be 0-999

--- End Test Case -------------------------------------

## test12

--- Test 12: setPortion() when invalid (high) ---------

Before

ID: 11

Name: test

Food Group: Undefined

Calories: 1

Test data: 1000

After

ID: 11

Name: test

Food Group: Undefined

Calories: 1

Error: java.lang.Exception: Calories must be 0-999

--- End Test Case -------------------------------------

# Measurement Tests

## test13

--- Test 13: Measurement Constructor when valid -----

Test data: cheese, 550, Unit.GRAMS

Measurement constructed

After

Ingredient: Cheese

Amount: 550

Units: g

--- End Test Case -----------------------------------

## test14

--- Test 14: Measurement Constructor when invalid ---

Test data: null, 0, null

Measurement constructor attempted

Error: java.lang.Exception:

Data could not be written

Reason(s):

java.lang.Exception: Ingredient cannot be null

java.lang.Exception: Amount must be 1-9999

java.lang.Exception: Unit cannot be null

--- End Test Case -----------------------------------

## test15

--- Test 15: setIngredient() when valid -------------

Before

Ingredient: Cheese

Amount: 100

Units: Undefined

Test data: milk

After

Ingredient: Milk

Amount: 100

Units: Undefined

--- End Test Case -----------------------------------

## test16

--- Test 16: setIngredient() when invalid -----------

Before

Ingredient: Cheese

Amount: 100

Units: Undefined

Test data: null

After

Ingredient: Cheese

Amount: 100

Units: Undefined

Error: java.lang.Exception: Ingredient cannot be null

--- End Test Case -----------------------------------

## test17

--- Test 17: setAmount() when valid -------------

Before

Ingredient: Cheese

Amount: 100

Units: Undefined

Test data: 550

After

Ingredient: Cheese

Amount: 550

Units: Undefined

--- End Test Case -----------------------------------

## test18

--- Test 18: setAmount() lowest possible --------------

Before

Ingredient: Cheese

Amount: 100

Units: Undefined

Test data: 1

After

Ingredient: Cheese

Amount: 1

Units: Undefined

--- End Test Case -----------------------------------

## test19

--- Test 19: setAmount() highest possible -------------

Before

Ingredient: Cheese

Amount: 100

Units: Undefined

Test data: 9999

After

Ingredient: Cheese

Amount: 9999

Units: Undefined

--- End Test Case -----------------------------------

## test20

--- Test 20: setAmount() when invalid (low) -------

Before

Ingredient: Cheese

Amount: 100

Units: Undefined

Test data: 0

After

Ingredient: Cheese

Amount: 100

Units: Undefined

Error: java.lang.Exception: Amount must be 1-9999

--- End Test Case -----------------------------------

## test21

--- Test 21: setAmount() when invalid (high) -------

Before

Ingredient: Cheese

Amount: 100

Units: Undefined

Test data: 10000

After

Ingredient: Cheese

Amount: 100

Units: Undefined

Error: java.lang.Exception: Amount must be 1-9999

--- End Test Case -----------------------------------

## test22

--- Test 22: setUnit() when valid --------------------

Before

Ingredient: Cheese

Amount: 100

Units: Undefined

Test data: Unit.GRAMS

After

Ingredient: Cheese

Amount: 100

Units: g

--- End Test Case -----------------------------------

--- Test 23: setUnit() when invalid -------------------

Before

Ingredient: Cheese

Amount: 100

Units: Undefined

Test data: null

After

Ingredient: Cheese

Amount: 100

Units: Undefined

Error: java.lang.Exception: Unit cannot be null

--- End Test Case -----------------------------------

# Recipe Tests

## test24

--- Test 24: Recipe Constructor when valid -----------

Test data: 'Garlic Bread', MealType.STARTER, 4, measurements, method

Recipe constructed

After:

-------------------------------------

0000 - Garlic Bread

Starter

4 Portion(s)

294.16 Calories per portion

Ingredients:

• White Bread - 152g

• Cheese - 100g

• Butter - 50g

• Garlic - 12g

Method:

1. Preheat oven to 180°C (350°F).

2. In a bowl, mix butter and minced garlic.

3. Spread the garlic butter on one side of each bread slice.

4. Sprinkle grated cheese on top and bake until the cheese is melted and bread is toasted.

-------------------------------------

--- End Test Case -------------------------------------

## test25

--- Test 25: Recipe Constructor when invalid ----------

Test data: '', null, 0, {}, {}

Recipe constructor attempted

Error: java.lang.Exception:

Data could not be written

Reason(s):

java.lang.Exception: Name cannot be blank or null

java.lang.Exception: MealType cannot be null

java.lang.Exception: Portions must be 1-99

java.lang.Exception: Measurements cannot be empty or null

java.lang.Exception: Method cannot be empty or null

--- End Test Case -------------------------------------

## test26

--- Test 26: setName() when valid ---------------------

Before:

-------------------------------------

0002 - test

Undefined

2 Portion(s)

0.26 Calories per portion

Ingredients:

• Apples - 1g

Method:

1. test method step

-------------------------------------

Test data: 'Updated Name'

After:

-------------------------------------

0002 - Updated Name

Undefined

2 Portion(s)

0.26 Calories per portion

Ingredients:

• Apples - 1g

Method:

1. test method step

-------------------------------------

--- End Test Case -------------------------------------

## test27

--- Test 27: setName() when invalid -------------------

Before:

-------------------------------------

0003 - test

Undefined

2 Portion(s)

0.26 Calories per portion

Ingredients:

• Apples - 1g

Method:

1. test method step

-------------------------------------

Test data: '123'

After:

-------------------------------------

0003 - test

Undefined

2 Portion(s)

0.26 Calories per portion

Ingredients:

• Apples - 1g

Method:

1. test method step

-------------------------------------

Error: java.lang.Exception: Name must be letters or characters: /&,.'-

--- End Test Case -------------------------------------

## test28

--- Test 28: setName() when null ----------------------

Before:

-------------------------------------

0004 - test

Undefined

2 Portion(s)

0.26 Calories per portion

Ingredients:

• Apples - 1g

Method:

1. test method step

-------------------------------------

Test data: ''

After:

-------------------------------------

0004 - test

Undefined

2 Portion(s)

0.26 Calories per portion

Ingredients:

• Apples - 1g

Method:

1. test method step

-------------------------------------

Error: java.lang.Exception: Name cannot be blank or null

--- End Test Case -------------------------------------

## test29

--- Test 29: setMealType() when valid -----------------

Before:

-------------------------------------

0005 - test

Undefined

2 Portion(s)

0.26 Calories per portion

Ingredients:

• Apples - 1g

Method:

1. test method step

-------------------------------------

Test data: MealType.MAIN

After:

-------------------------------------

0005 - test

Main

2 Portion(s)

0.26 Calories per portion

Ingredients:

• Apples - 1g

Method:

1. test method step

-------------------------------------

--- End Test Case -------------------------------------

## test30

--- Test 30: setMealType() when invalid ---------------

Before:

-------------------------------------

0006 - test

Undefined

2 Portion(s)

0.26 Calories per portion

Ingredients:

• Apples - 1g

Method:

1. test method step

-------------------------------------

Test data: null

After:

-------------------------------------

0006 - test

Undefined

2 Portion(s)

0.26 Calories per portion

Ingredients:

• Apples - 1g

Method:

1. test method step

-------------------------------------

Error: java.lang.Exception: MealType cannot be null

--- End Test Case -------------------------------------

## test31

--- Test 31: setPortion() when valid ------------------

Before:

-------------------------------------

0007 - test

Undefined

2 Portion(s)

0.26 Calories per portion

Ingredients:

• Apples - 1g

Method:

1. test method step

-------------------------------------

Test data: 5

After:

-------------------------------------

0007 - test

Undefined

5 Portion(s)

0.10 Calories per portion

Ingredients:

• Apples - 1g

Method:

1. test method step

-------------------------------------

--- End Test Case -------------------------------------

## test32

--- Test 32: setPortion() lowest possible -------------

Before:

-------------------------------------

0008 - test

Undefined

2 Portion(s)

0.26 Calories per portion

Ingredients:

• Apples - 1g

Method:

1. test method step

-------------------------------------

Test data: 1

After:

-------------------------------------

0008 - test

Undefined

1 Portion(s)

0.52 Calories per portion

Ingredients:

• Apples - 1g

Method:

1. test method step

-------------------------------------

--- End Test Case --------------------------------------

## test33

--- Test 33: setPortion() highest possible -------------

Before:

-------------------------------------

0009 - test

Undefined

2 Portion(s)

0.26 Calories per portion

Ingredients:

• Apples - 1g

Method:

1. test method step

-------------------------------------

Test data: 99

After:

-------------------------------------

0009 - test

Undefined

99 Portion(s)

0.01 Calories per portion

Ingredients:

• Apples - 1g

Method:

1. test method step

-------------------------------------

--- End Test Case -------------------------------------

## test34

--- Test 34: setPortion() when invalid (low) ----------

Before:

-------------------------------------

0010 - test

Undefined

2 Portion(s)

0.26 Calories per portion

Ingredients:

• Apples - 1g

Method:

1. test method step

-------------------------------------

Test data: 0

After:

-------------------------------------

0010 - test

Undefined

2 Portion(s)

0.26 Calories per portion

Ingredients:

• Apples - 1g

Method:

1. test method step

-------------------------------------

Error: java.lang.Exception: Portions must be 1-99

--- End Test Case -------------------------------------

## test35

--- Test 35: setPortion() when invalid (high) ---------

Before:

-------------------------------------

0011 - test

Undefined

2 Portion(s)

0.26 Calories per portion

Ingredients:

• Apples - 1g

Method:

1. test method step

-------------------------------------

Test data: 100

After:

-------------------------------------

0011 - test

Undefined

2 Portion(s)

0.26 Calories per portion

Ingredients:

• Apples - 1g

Method:

1. test method step

-------------------------------------

Error: java.lang.Exception: Portions must be 1-99

--- End Test Case -------------------------------------

## test36

--- Test 36: setMeasurements() when valid -------------

Before:

-------------------------------------

0012 - test

Undefined

2 Portion(s)

0.26 Calories per portion

Ingredients:

• Apples - 1g

Method:

1. test method step

-------------------------------------

Test data: newMeasurements = {(milk,500, Unit.MILLILITRES), (cheese, 225, Unit.GRAMS)}

After:

-------------------------------------

0012 - test

Undefined

2 Portion(s)

557.25 Calories per portion

Ingredients:

• Milk - 500ml

• Cheese - 225g

Method:

1. test method step

-------------------------------------

--- End Test Case -------------------------------------

## test37

--- Test 37: setMeasurements() when invalid -----------

Before:

-------------------------------------

0013 - test

Undefined

2 Portion(s)

0.26 Calories per portion

Ingredients:

• Apples - 1g

Method:

1. test method step

-------------------------------------

Test data: null

After:

-------------------------------------

0013 - test

Undefined

2 Portion(s)

0.26 Calories per portion

Ingredients:

• Apples - 1g

Method:

1. test method step

-------------------------------------

Error: java.lang.Exception: Measurements cannot be empty or null

--- End Test Case -------------------------------------

## test38

--- Test 38: setMethod() when valid -------------------

Before:

-------------------------------------

0014 - test

Undefined

2 Portion(s)

0.26 Calories per portion

Ingredients:

• Apples - 1g

Method:

1. test method step

-------------------------------------

Test data: newMethod= {'Step 1','Step 2','Step 3'}

After:

-------------------------------------

0014 - test

Undefined

2 Portion(s)

0.26 Calories per portion

Ingredients:

• Apples - 1g

Method:

1. Step 1

2. Step 2

3. Step 3

-------------------------------------

--- End Test Case -------------------------------------

## test39

--- Test 39: setMethod() when invalid -----------------

Before:

-------------------------------------

0015 - test

Undefined

2 Portion(s)

0.26 Calories per portion

Ingredients:

• Apples - 1g

Method:

1. test method step

-------------------------------------

Test data: null

After:

-------------------------------------

0015 - test

Undefined

2 Portion(s)

0.26 Calories per portion

Ingredients:

• Apples - 1g

Method:

1. test method step

-------------------------------------

Error: java.lang.Exception: Method cannot be empty or null

--- End Test Case -------------------------------------

## test40

--- Test 40: addMeasurement() when valid -------------------

Before:

-------------------------------------

0016 - test

Undefined

2 Portion(s)

0.26 Calories per portion

Ingredients:

• Apples - 1g

Method:

1. test method step

-------------------------------------

Test data: newMeasurement = (cheese, 350, Unit.GRAMS)

After:

-------------------------------------

0016 - test

Undefined

2 Portion(s)

703.76 Calories per portion

Ingredients:

• Apples - 1g

• Cheese - 350g

Method:

1. test method step

-------------------------------------

--- End Test Case -------------------------------------

## test41

--- Test 41: addMeasurement() when invalid -----------------

Before:

-------------------------------------

0017 - test

Undefined

2 Portion(s)

0.26 Calories per portion

Ingredients:

• Apples - 1g

Method:

1. test method step

-------------------------------------

Test data: null

After:

-------------------------------------

0017 - test

Undefined

2 Portion(s)

0.26 Calories per portion

Ingredients:

• Apples - 1g

Method:

1. test method step

-------------------------------------

Error: java.lang.Exception: Measurement cannot be null

--- End Test Case -------------------------------------

## test42

--- Test 42: addMethod() when valid -------------------

Before:

-------------------------------------

0018 - test

Undefined

2 Portion(s)

0.26 Calories per portion

Ingredients:

• Apples - 1g

Method:

1. test method step

-------------------------------------

Test data: 'Combine the egg whites and the flour'

After:

-------------------------------------

0018 - test

Undefined

2 Portion(s)

0.26 Calories per portion

Ingredients:

• Apples - 1g

Method:

1. test method step

2. Combine the egg whites and the flour

-------------------------------------

--- End Test Case -------------------------------------

## test43

--- Test 43: addMethod() shortest possible ------------

Before:

-------------------------------------

0019 - test

Undefined

2 Portion(s)

0.26 Calories per portion

Ingredients:

• Apples - 1g

Method:

1. test method step

-------------------------------------

Test data: 'a'

After:

-------------------------------------

0019 - test

Undefined

2 Portion(s)

0.26 Calories per portion

Ingredients:

• Apples - 1g

Method:

1. test method step

2. a

-------------------------------------

--- End Test Case -------------------------------------

## test44

--- Test 44: addMethod() longest possible -------------

Before:

-------------------------------------

0020 - test

Undefined

2 Portion(s)

0.26 Calories per portion

Ingredients:

• Apples - 1g

Method:

1. test method step

-------------------------------------

Test data: 'aaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaa'

After:

-------------------------------------

0020 - test

Undefined

2 Portion(s)

0.26 Calories per portion

Ingredients:

• Apples - 1g

Method:

1. test method step

2. aaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaa

-------------------------------------

--- End Test Case -------------------------------------

## test45

--- Test 45: addMethod() when invalid (short) ---------

Before:

-------------------------------------

0021 - test

Undefined

2 Portion(s)

0.26 Calories per portion

Ingredients:

• Apples - 1g

Method:

1. test method step

-------------------------------------

Test data: ''

After:

-------------------------------------

0021 - test

Undefined

2 Portion(s)

0.26 Calories per portion

Ingredients:

• Apples - 1g

Method:

1. test method step

-------------------------------------

Error: java.lang.Exception: Step must be between 0 and 100 characters

--- End Test Case -------------------------------------

## test46

--- Test 46: addMethod() when invalid (long) ----------

Before:

-------------------------------------

0022 - test

Undefined

2 Portion(s)

0.26 Calories per portion

Ingredients:

• Apples - 1g

Method:

1. test method step

-------------------------------------

Test data: 'aaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaa'

After:

-------------------------------------

0022 - test

Undefined

2 Portion(s)

0.26 Calories per portion

Ingredients:

• Apples - 1g

Method:

1. test method step

-------------------------------------

Error: java.lang.Exception: Step must be between 0 and 100 characters

--- End Test Case -------------------------------------

## test47

--- Test 47: totalCalories() --------------------------

Before:

-------------------------------------

0023 - Garlic Bread

Starter

4 Portion(s)

296.14 Calories per portion

Ingredients:

• White Bread - 155g

• Cheese - 100g

• Butter - 50g

• Garlic - 12g

Method:

1. Preheat oven to 180°C (350°F).

2. In a bowl, mix butter and minced garlic.

3. Spread the garlic butter on one side of each bread slice.

4. Sprinkle grated cheese on top and bake until the cheese is melted and bread is toasted.

-------------------------------------

Returned: 296.1425

--- End Test Case -------------------------------------

# WeeklyMenu Tests

## test48

--- Test 48: addRecipe() when valid -------

Before:

Day Starter Main Dessert

Monday ----- ----- -----

Tuesday ----- ----- -----

Wednesday ----- ----- -----

Thursday ----- ----- -----

Friday ----- ----- -----

Test data: Garlic Bread, 3, 3

After:

Day Starter Main Dessert

Monday ----- ----- -----

Tuesday ----- ----- -----

Wednesday ----- ----- Garlic Bread

Thursday ----- ----- -----

Friday ----- ----- -----

--- End Test Case -------------------------

## test49

--- Test 49: addRecipe() when invalid -----

Before:

Day Starter Main Dessert

Monday ----- ----- -----

Tuesday ----- ----- -----

Wednesday ----- ----- -----

Thursday ----- ----- -----

Friday ----- ----- -----

Test data: null, 6, 0

After:

Day Starter Main Dessert

Monday ----- ----- -----

Tuesday ----- ----- -----

Wednesday ----- ----- -----

Thursday ----- ----- -----

Friday ----- ----- -----

Error Message:

'java.lang.Exception:

Errors:

Recipe cannot be null

Day must be 1-5

Course must be 1-3'

--- End Test Case --------------------------

## test50

--- Test 50: removeRecipe() when valid -----

Before:

Day Starter Main Dessert

Monday Garlic Bread Potato Leek So- Apple and Cinn-

Tuesday Stuffed Garlic- Cheesy Baked P- Cheese and App-

Wednesday Lettuce Wraps - Ground Beef an- Cinnamon Apple-

Thursday Tomato and Lee- Apple and Jam - Apple Jam Thum-

Friday Celery and Che- Egg and Onion - Cheese and Jam-

Test data: 3, 3

After:

Day Starter Main Dessert

Monday Garlic Bread Potato Leek So- Apple and Cinn-

Tuesday Stuffed Garlic- Cheesy Baked P- Cheese and App-

Wednesday Lettuce Wraps - Ground Beef an- -----

Thursday Tomato and Lee- Apple and Jam - Apple Jam Thum-

Friday Celery and Che- Egg and Onion - Cheese and Jam-

--- End Test Case --------------------------

## test51

--- Test 51: removeRecipe() when invalid ---

Before:

Day Starter Main Dessert

Monday Garlic Bread Potato Leek So- Apple and Cinn-

Tuesday Stuffed Garlic- Cheesy Baked P- Cheese and App-

Wednesday Lettuce Wraps - Ground Beef an- Cinnamon Apple-

Thursday Tomato and Lee- Apple and Jam - Apple Jam Thum-

Friday Celery and Che- Egg and Onion - Cheese and Jam-

Test data: 6, 0

After:

Day Starter Main Dessert

Monday Garlic Bread Potato Leek So- Apple and Cinn-

Tuesday Stuffed Garlic- Cheesy Baked P- Cheese and App-

Wednesday Lettuce Wraps - Ground Beef an- Cinnamon Apple-

Thursday Tomato and Lee- Apple and Jam - Apple Jam Thum-

Friday Celery and Che- Egg and Onion - Cheese and Jam-

Error Message:

'java.lang.Exception: Errors:

Day must be 1-5

Course must be 1-3'

--- End Test Case --------------------------

## test52

--- Test 52: clearMenu() -------------------

Before:

Day Starter Main Dessert

Monday Garlic Bread Potato Leek So- Apple and Cinn-

Tuesday Stuffed Garlic- Cheesy Baked P- Cheese and App-

Wednesday Lettuce Wraps - Ground Beef an- Cinnamon Apple-

Thursday Tomato and Lee- Apple and Jam - Apple Jam Thum-

Friday Celery and Che- Egg and Onion - Cheese and Jam-

After:

Day Starter Main Dessert

Monday ----- ----- -----

Tuesday ----- ----- -----

Wednesday ----- ----- -----

Thursday ----- ----- -----

Friday ----- ----- -----

--- End Test Case --------------------------

## test53

--- Test 53: isEmpty() when empty -----------

Before:

Day Starter Main Dessert

Monday ----- ----- -----

Tuesday ----- ----- -----

Wednesday ----- ----- -----

Thursday ----- ----- -----

Friday ----- ----- -----

Returned:

true

--- End Test Case --------------------------

## test54

--- Test 54: isEmpty() when not empty--------

Before:

Day Starter Main Dessert

Monday ----- ----- -----

Tuesday ----- ----- -----

Wednesday ----- ----- Garlic Bread

Thursday ----- ----- -----

Friday ----- ----- -----

Returned:

false

--- End Test Case --------------------------

# Utility Tests

## test55

--- Test 55: isName() when valid --------

Test data: 'Hannah's Potato & Leek Soup'

Returned: true

--- End Test Case -----------------------

## test56

--- Test 56: isName() when invalid ------

Test data: '123!\*%abc'

Returned: false

--- End Test Case -----------------------

## test57

--- Test 57: isNumeric() when valid -----

Test data: '1536'

Returned: false

--- End Test Case -----------------------

## test58

--- Test 58: isNumeric() when invalid ---

Test data: '123!\*%abc'

Returned: false

--- End Test Case -----------------------

## test59

--- Test 59: isMeasurement() when valid -

Test data: '400 g'

Returned: true

--- End Test Case -----------------------

## test60

--- Test 60: isMeasurement() when invalid ---

Test data: '123!\*%abc'

Returned: false

--- End Test Case ------------------