

Project Document

Meeting Minutes:

First:

Minutes of the first meeting^{6,4}
Date: 11th March^{6,4}
Time: 21:30^{6,4}
Venue: Zoom^{6,4}
Attending: Keith, Steven, Anyia^{6,4}
Recorder: Keith^{6,4}
^{6,4}
Agenda:^{6,4}
1. Setting up the team repository^{6,4}
2. Team formation form^{6,4}
3. Task allocation^{6,4}
4. UML diagram and attributes tables^{6,4}
^{6,4}
Meeting Content:^{6,4}
1.Setting up the team repository^{6,4}
1.1 The team repository was set by us together in the meeting as according of the specification.^{6,4}
1.2 Keith invited the TAs to join the repository.^{6,4}
1.3 Anyia and Steven created the branches.^{6,4}
2 Problem for accessing the branches by Anyia and Steven^{6,4}
2.1 Anyia and Steven could not access the branches after creating the branches.^{6,4}
2.2 The problem is solved after Keith change the setting of the repository.^{6,4}
3 Team formation form^{6,4}
3.1 Keith will become the team leader.^{6,4}
3.2 The allocated task:^{6,4}

Group member ^{6,3}	The task allocated ^{6,3}	^{6,3}
Anyia ^{6,3}	Function A ^{6,3}	^{6,3}
Steven ^{6,3}	Function B ^{6,3}	^{6,3}
Keith ^{6,3}	Function C ^{6,3}	^{6,3}

3.3 The team formation form was completed in the meeting and added in the readme.md file.^{6,4}
4 Problem of adding the form into the readme.md file^{6,4}
4.1 The team formation form can not be added directly in form of excel file.^{6,4}
4.2 The problem is solved by writing the form by html code.^{6,4}
5 The UML diagram and attributes tables^{6,4}
5.1 The diagram and attributes table will be done in the next meeting.^{6,4}
5.2 Everyone need to read the specification and think about it before the next

meeting.^{6,4}
6 The meeting adjournment and the next meeting^{6,4}
6.1 The meeting was adjourned at 23:00 and the next meeting will be held at the 14th March after the end of the COMP3111 lecture.^{6,4}
6.2 Steven booked the room 353 in the library for the next meeting.^{6,4}
^{6,4}

Second:

Minutes of the second meeting^{4,1}

Date: 18th March^{4,1}

Time: 13:30^{4,1}

Venue: Zoom^{4,1}

Attending: Keith, Steven, Anyia^{4,1}

Recorder: Keith^{4,1}

^{4,1}

Agenda:^{4,1}

1. Attributes table^{4,1}

2. UML diagram^{4,1}

^{4,1}

Meeting content:^{4,1}

1 Attributes table^{4,1}

1.1 Everyone spent 30 minutes to extract the attributes of the object from the specification and make the attributes table by themselves.^{4,1}

1.2 Every groupmates shared their own attributes tables and had a discussion on them.^{4,1}

1.3 Most of attributes of objects in the attributes table were filled excepted the attributes of the shipment and the staff.^{4,1}

2 The problem on the definition of the shipment and staff^{4,1}

2.1 Everyone were confused about whether the delivery notes are referring to the shipment and whether the staff need to be further classified. ^{4,1}

2.2 The problem is solved by send the email to the TA.^{4,1}

2.3 Since further clarification is needed, the discussion on the attributes table will be continued in the next meeting.^{4,1}

3 Allocated task:^{4,1}

3.1 Due to the time was running short, The UML diagram will be discussed in the next meeting.^{4,1}

3.2 Everyone need to prepare for the UML diagram for the next meeting.^{4,1}

4 The meeting adjournment and the next meeting^{4,1}

4.1 The meeting was adjourned at 23:00 and the next meeting will be held at the 20th March at the 15:00.^{4,1}

4.2 Keith booked the LC3 in the library learning common for the next meeting.^{4,1}

^{4,1}

Third:

Minutes of the third meeting^{4,4}

Date: 20th March^{4,4}

Time: 15:00^{4,4}

Venue: LC3, Library Learning Common^{4,4}

Attending: Keith, Anyia, Steven^{4,4}

Recorder: Keith^{4,4}

Agenda: ^{4,4}

1. Follow-up of the attributes table^{4,4}
2. UML design^{4,4}

Meeting Content:^{4,4}

- 1 The follow-up of the attributes table^{4,4}
 - 1.1 The TA replied the questions about the definition of the shipment and the staff.^{4,4}
 - 1.2 The TA said the type of the staff does not need to be defined.^{4,4}
 - 1.3 The TA said the delivery notes is referring to the shipment.^{4,4}
- 2 The attributes table^{4,4}
 - 2.1 The attributes of the staff and the shipment was completed in the meeting.^{4,4}
 - 2.2 Everyone spent 15 minutes to proofread the attributes table.^{4,4}
 - 2.3 The attributes table was finalized in the meeting.^{4,4}
- 3 UML diagram^{4,4}
 - 3.1 Everyone spent 20 minutes on drawing the UML diagram by themselves.^{4,4}
 - 3.2 Everyone shared their own work and had a discussion on the UML diagram.^{4,4}
 - 3.3 A conclusion had drawn and summarized into one diagram.^{4,4}
- 4 The allocation task:^{4,4}
 - 4.1 Keith will finalize the UML diagram and send to Steven and Anyia for proofread.^{4,4}
 - 4.2 The deadline of the task is 21st March.^{4,4}
- 5 The meeting adjournment and the next meeting^{4,4}
 - 5.1 The meeting was adjourned at 17:00 and the next meeting will be held after the activity 2 starts.^{4,4}

Fourth:

Minutes of the fourth meeting^{4,1}

Date: 1st April^{4,1}

Time: 22:35^{4,1}

Venue: Discord^{4,1}

Attending: Keith, ~~Avia~~, Steven^{4,1}

Recording: Keith^{4,1}

Agenda:^{4,1}

1. Discussion on the code^{4,1}
2. Discussion on the allocation of task^{4,1}

Meeting:^{4,1}

- 1 Discussion on the code^{4,1}
 - 1.1 Everyone agreed to use the skeleton code to speed up the start of software development.^{4,1}
 - 1.2 The skeleton code tried to be imported in the meeting.^{4,1}
 - 1.3 Problem was found when importing the skeleton code.^{4,1}
- 2 Problem of importing the skeleton code^{4,1}
 - 2.1 The skeleton code can not be imported as a ~~gradle~~ project.^{4,1}
 - 2.2 To solve the problem, ~~Avia~~ checked the FAQ document that it is allowed to import the skeleton code as Maven project.^{4,1}
 - 2.3 The problem is solved after importing the Maven project.^{4,1}
- 3 The allocation of task^{4,1}
 - 3.1 Keith will create the template of each scene to unify the GUI design of the program.^{4,1}
 - 3.2 ~~Avia~~ and Steven will design a class to solve the optimization of the calculation.^{4,1}
 - 3.3 Steven will do the grant chart.^{4,1}
 - 3.4 ~~Avia~~ will do the burnt down chart.^{4,1}
 - 3.5 Keith will do the integration of the program after the every function is finished by each member.^{4,1}
- 4 The meeting adjournment and next meeting^{4,1}
 - 4.1 The meeting was adjourned at 23:59 and the next meeting will be held on 13th April.^{4,1}

Fifth:

Minutes of the fifth meeting

Date: 1st May

Time: 13:00

Venue: Discord

Attending: Keith, Anyia, Steven

Recording: Keith

Agenda:

1. Follow up on the progress

2. Timeslot of the demo

3. Review of the program

4. Unit testing and coverage

5. Java Documentation

6. Integration of the program

Meeting Content:

1 Follow up on the progress

1.1 The burn down chart and the grant chart is done.

1.2 Most of the group mates had finished 70% of the program

2 The timeslot of the demo

2.1 The timeslot of the demo is decide and Keith will be the representative of group 19 in the demo.

3 Review of the program

3.1 The master branch that created using the Eclipse is deleted in the meeting

4 Unit test

4.1 The unit testing for each member will be done independently by each member since members are familiar with their own work

4.2 Unit testing for each function should be more than 65%.

5 The Java Documentation

5.1 The documentation of each function will be done by each member independently.

6 Integration of the program

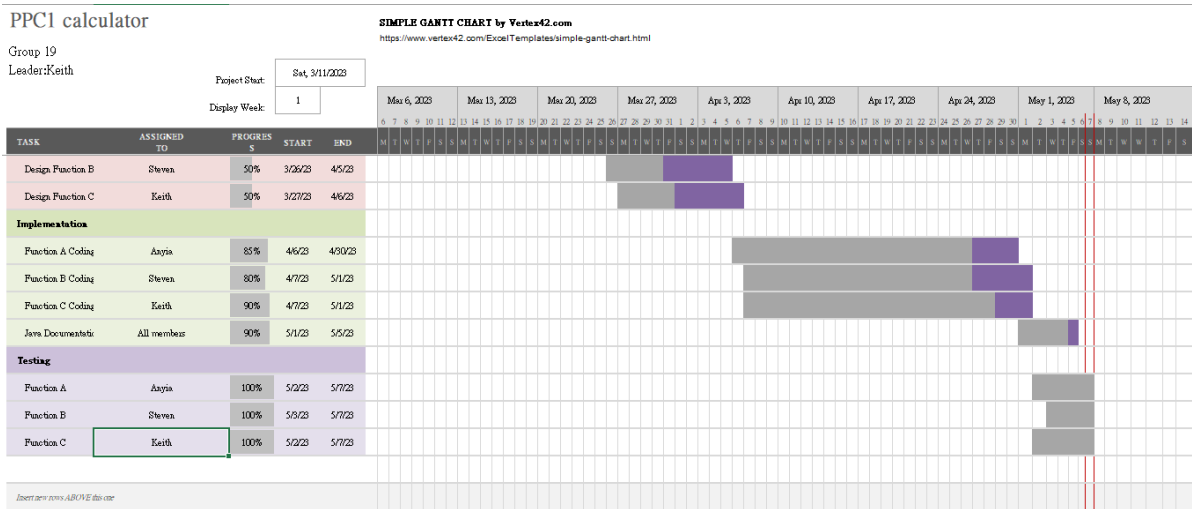
6.1 Keith will finalize the program.

6.2 Unit testing report, coverage report, and the java documentation will be generated by Keith.

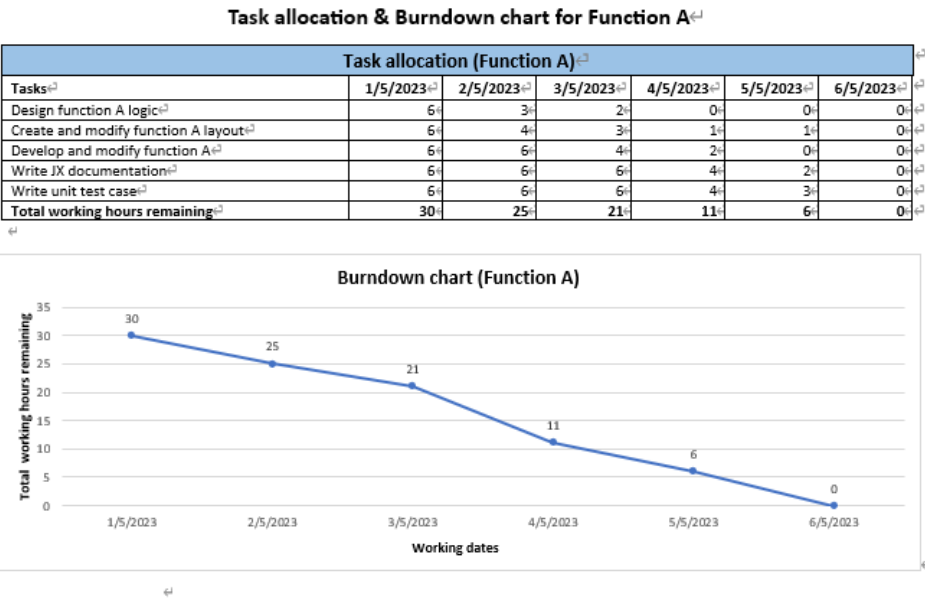
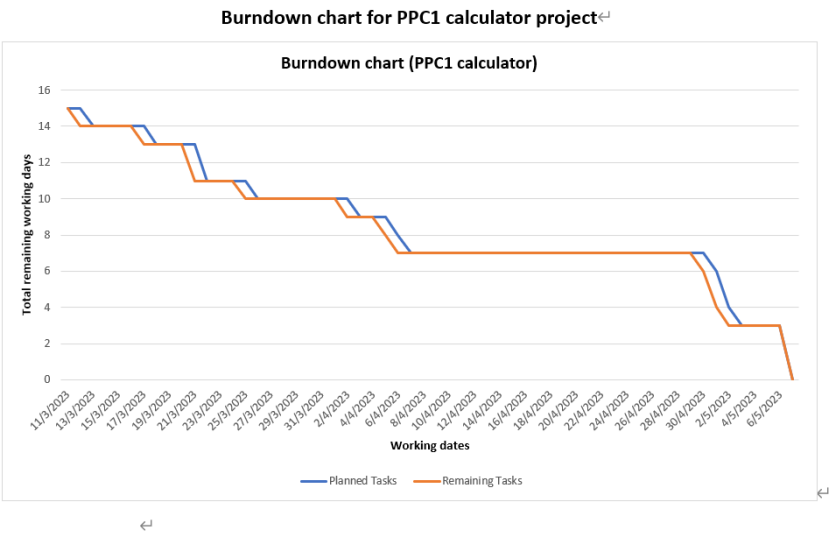
7 The meeting adjournment

7.1 The meeting was adjourned at 13:55.

Gantt Chart:

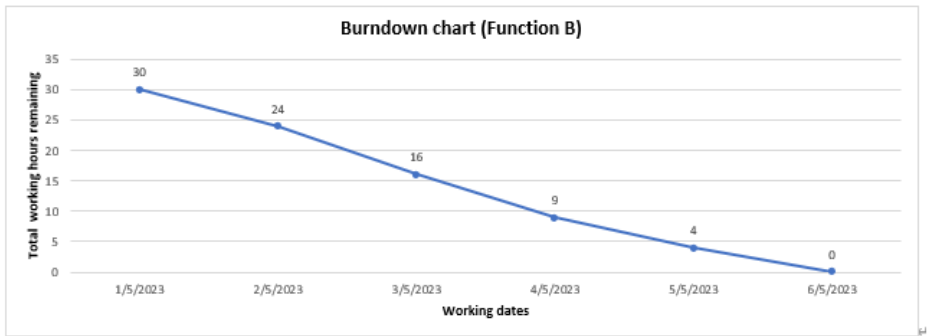


Burndown Chart:



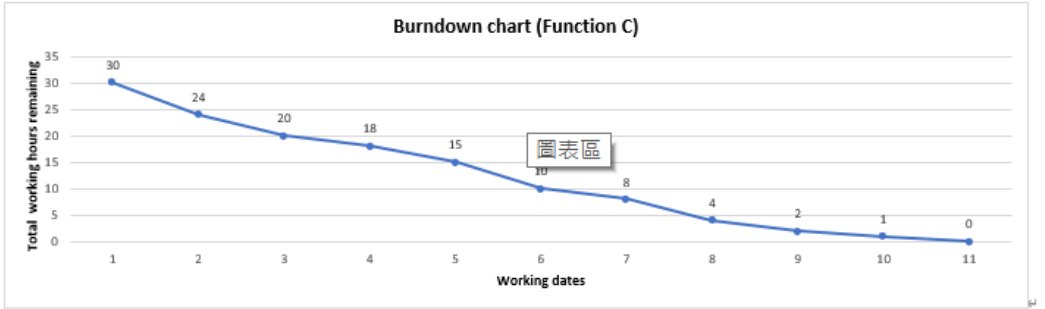
Task allocation & Burndown chart for Function B

Task allocation (Function B)						
Tasks	1/5/2023	2/5/2023	3/5/2023	4/5/2023	5/5/2023	6/5/2023
Design function B logic	6	3	0	0	0	0
Create and modify function B layout	6	4	2	0	0	0
Develop and modify function B	6	5	3	2	0	0
Write JX documentation	6	6	5	3	2	0
Write unit test case	6	6	6	4	2	0
Total working hours remaining	30	24	16	9	4	0



Task allocation & Burndown chart for Function C

Task allocation (Function C)											
Tasks	26/4/20 23	27/4/20 23	28/4/20 23	29/4/20 23	30/4/20 23	1/5/20 23	2/5/20 23	3/5/20 23	4/5/20 23	5/5/20 23	6/5/20 23
Design function C logic	6	2	0	0	0	0	0	0	0	0	0
Create and modify function C layout	6	4	3	2	1	0	0	0	0	0	0
Develop and modify function C	6	6	5	4	2	0	0	0	0	0	0
Write JX documentation	6	6	6	6	6	4	2	1	0	0	0
Write unit test case	6	6	6	6	6	6	6	3	2	1	0
Total working hours remaining	30	24	20	18	15	10	8	4	2	1	0



Git Commit Log:

Update the JavaDoc ikeith0387 committed 20 minutes ago		22fea71	
Upload Gantt Chart ikeith0387 committed 50 minutes ago		84d4e71	
Modify the directory path of test files ikeith0387 committed 1 hour ago		2fa8c7d	
Remove the prompt test in the GUI of all functions ikeith0387 committed 1 hour ago		d705ac9	
Create COMP3111_GP19_TaskAllocation_And_BurndownChart.docx ikeith0387 committed 1 hour ago		ae2c77e	
Merge remote-tracking branch 'origin/Function_B' ikeith0387 committed 1 hour ago		c951c2c	
updated java doc copycatv committed 1 hour ago		f3321cd	
Update the class after renaming ikeith0387 committed 1 hour ago		0878e27	
Update the classes ikeith0387 committed 1 hour ago		eb8c64c	
Rename the class to distinguish which function do they belongs to. ikeith0387 committed 1 hour ago		7101025	
Merge remote-tracking branch 'origin/Function_A' ikeith0387 committed 1 hour ago		0a336ca	
Create COMP3111_Project_GP19_BurndownChart.xlsx ikeith0387 committed 1 hour ago		25aadad	

Filters

ispr isclosed

Labels 0

Milestones 0

New pull request

Clear current search query, filters, and sorts

☐

0 Open

4 Closed

Author

Label

Projects

Milestones

Reviews

Assignee

Sort

☐

Merge Function c

#4 by ikeith0387 was merged yesterday

☐

Merge Function a

#3 by ikeith0387 was merged yesterday

☐

Combine Function A into main branch

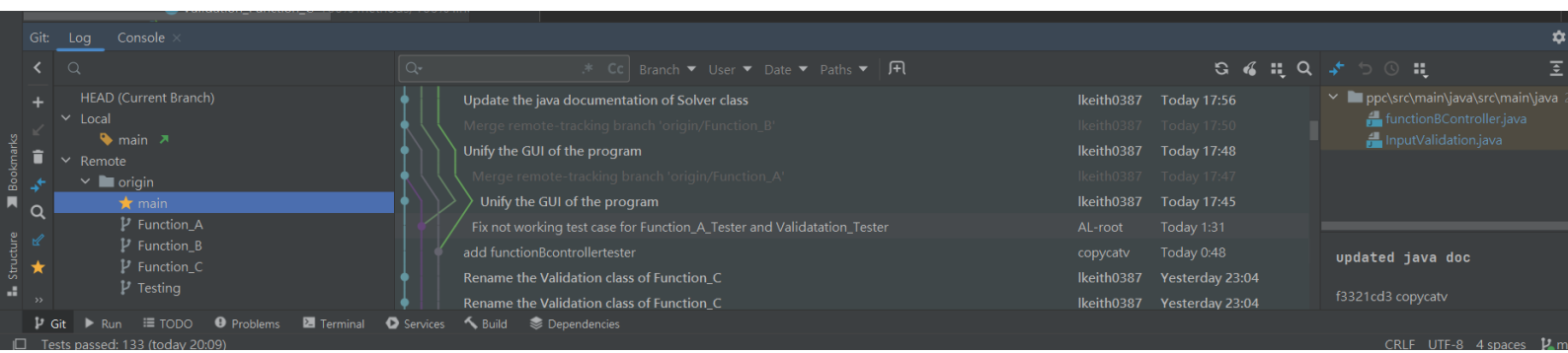
#2 by ikeith0387 was closed yesterday

☐

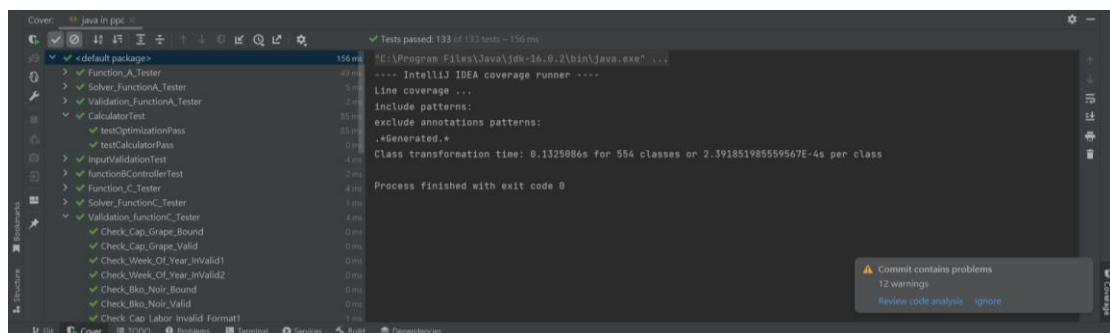
clone the project to the Function_C branch

#1 by ikeith0387 was merged 2 weeks ago

ProTip! Click a checkbox on the left to edit multiple issues at once.



Unit Testing:



Coverage Report:

Coverage: java in ppc			
Element	Class, %	Method, %	Line, %
src	68% (11/16)	67% (41/61)	76% (324/426)
main	68% (11/16)	67% (41/61)	76% (324/426)
java	68% (11/16)	67% (41/61)	76% (324/426)
Function_A	80% (4/5)	75% (15/20)	81% (108/133)
function_B	100% (3/3)	73% (11/15)	68% (86/125)
Function_C	80% (4/5)	75% (15/20)	83% (130/156)
AppMain	0% (0/1)	0% (0/1)	0% (0/1)
comp3111pcc1Controller	0% (0/1)	0% (0/4)	0% (0/7)
Main	0% (0/1)	0% (0/1)	0% (0/4)

JavaDoc:

OVERVIEWPACKAGECLASSTREEINDEXHELP

SUMMARY: NESTED | FIELD | CONSTR | METHODDETAIL: FIELD | CONSTR | METHOD

SEARCH:

Package src.main.java.Function_A

Class Function_A

java.lang.Object[Ⓜ]
src.main.java.Function_A.Function_A

public class **Function_A**

extends Object[Ⓜ]

Function_A class is used to perform compute the optimal mix of wines to be produced in the year to maximize the gross profit of the winery

Constructor Summary

Constructors

Constructor	Description
Function_A ()	

Method Summary

All MethodsStatic MethodsInstance MethodsConcrete Methods

Modifier and Type	Method	Description
javafx.collections.ObservableList<String>	Data_Validation (String [Ⓜ] [] input)	Validates the data input
void	Get_Data (String [Ⓜ] [] input)	Converts an array of input into numeric data and stores in the fields of the class.
Result_Function_A	Get_Result ()	Calculates the optimal volume of two wines (Ros [Ⓜ] and P-Noir), then used them to calculate VCL, Sales Revenue, Gross Profit and Profit Margin.
javafx.collections.ObservableList<String>	Get_Warning_Message (int Opt_Rose, int Opt_Noir)	Checks whether the optimal solution of the volume of Ros [Ⓜ] and P-Noir will lead to abnormal situation and produces warning messages for abnormal situations.
static double	roundTwoSigFig (double value, int places)	

Methods inherited from class java.lang.Object[Ⓜ]

clone[Ⓜ], equals[Ⓜ], finalize[Ⓜ], getClass[Ⓜ], hashCode[Ⓜ], notify[Ⓜ], notifyAll[Ⓜ], toString[Ⓜ], wait[Ⓜ], wait[Ⓜ], wait[Ⓜ]

Constructor Details

Function_A

public Function_A()

Method Details

Get_Result

OVERVIEWPACKAGECLASSTREEINDEXHELP

SUMMARY: NESTED | FIELD | CONSTR | METHODDETAIL: FIELD | CONSTR | METHOD

SEARCH:

Package src.main.java.function_B

Class functionBController

java.lang.Object[Ⓜ]
src.main.java.function_B.functionBController

public class **functionBController**

extends Object[Ⓜ]

A Controller for function B in ppei calculator project.

Constructor Summary

Constructors

Constructor	Description
functionBController ()	

Method Summary

All MethodsInstance MethodsConcrete Methods

Modifier and Type	Method	Description
javafx.collections.ObservableList<String>	datainvalid (String [Ⓜ] [] input)	A function to call an extra class to verify if the input are in valid formats.
void	Display_Message (javafx.collections.ObservableList<String> Message)	display warning messages
void	Input_Data (String [Ⓜ] [] input)	to get user input in integer or float.
javafx.collections.ObservableList<String>	message (int opt_Rose, int opt_Noir)	to get warning messages
void	Output_Data (int opt_Rose, int opt_Noir, int opt_total, int maxRevenue, int sur_labor, int sur_grape, int sur_total)	Output text in textfields.
javafx.collections.ObservableList<String>	surplus_message (int sur_labor, int sur_grape)	to get error messages of condition when consumption greater than capacity
void	toclickB (javafx.event.ActionEvent event)	Handles mouse click event on "run" button
void	toexitB (javafx.event.ActionEvent event)	Handle mouse click event on "Exit" button

Methods inherited from class java.lang.Object[Ⓜ]

clone[Ⓜ], equals[Ⓜ], finalize[Ⓜ], getClass[Ⓜ], hashCode[Ⓜ], notify[Ⓜ], notifyAll[Ⓜ], toString[Ⓜ], wait[Ⓜ], wait[Ⓜ], wait[Ⓜ]

Package src.main.java.Function_C

Class Function_C

java.lang.Object[Ⓜ]
src.main.java.Function_C.Function_C

public class Function_C
extends Object[Ⓜ]

The Function_C class contains the process performed in the function C.

Constructor Summary

Constructors	
Constructor	Description
Function_C()	

Method Summary

All Methods	Instance Methods	Concrete Methods	
Modifier and Type	Method		Description
javaafx.collections.ObservableList<String [Ⓜ] >	Data_Validation(String [Ⓜ] [] input)		Validates the data input
void	Get_Data(String [Ⓜ] [] input)		Converts an array of the texted input into numeric data and stores in the fields of the class.
Result_Function_C	Get_Result()		Calculates the optimal solution of the volume of Ros [Ⓜ] and P-Noir with or without any backorder of Ros [Ⓜ] and P-Noir to maximize the revenue
javaafx.collections.ObservableList<String [Ⓜ] >	Get_Warning_Message(int Opt_Rose, int Opt_Noir)		Checks whether the optimal solution of the volume of Ros [Ⓜ] and P-Noir will lead to abnormal situation and produces warning messages for abnormal situations.
Methods inherited from class java.lang.Object [Ⓜ]			
clone [Ⓜ] , equals [Ⓜ] , finalize [Ⓜ] , getClass [Ⓜ] , hashCode [Ⓜ] , notify [Ⓜ] , notifyAll [Ⓜ] , toString [Ⓜ] , wait [Ⓜ] , wait [Ⓜ] , wait [Ⓜ]			

Constructor Details

Function_C
public Function_C()

Package src.main.java.Function_C

Class Solver_Function_C

java.lang.Object[Ⓜ]
src.main.java.Function_C.Solver_Function_C

public class Solver_Function_C
extends Object[Ⓜ]

The Solver_Function_C class provides solutions for calculation of some outputs in functions, including optimal volume of Ros[Ⓜ], optimal volume of P-Noir, and the revenue obtain with specified volume of Ros[Ⓜ] and P-Noir.

Constructor Summary

Constructors	
Constructor	Description
Solver_Function_C(int Cap_Labor, int Cap_Grape, float Prc_Rose, float Prc_Noir)	Initializes a new Solver_Function_C object that contains information on the capacity of Labor and Grape and the price of Ros [Ⓜ] and P-Noir.

Method Summary

All Methods	Instance Methods	Concrete Methods	
Modifier and Type	Method		Description
float	Calculation(int Mun_Rose, int Mun_Noir)		Calculates the revenue obtained with specified volume of Ros [Ⓜ] and P-Noir.
int[]	Opt_Solution(boolean celling, int celling_of_Rose, int celling_of_Noir)		Calculates the optimal solution of volume of Ros [Ⓜ] and P-Noir to obtain the highest revenue and store the result into an int array.
Methods inherited from class java.lang.Object [Ⓜ]			
clone [Ⓜ] , equals [Ⓜ] , finalize [Ⓜ] , getClass [Ⓜ] , hashCode [Ⓜ] , notify [Ⓜ] , notifyAll [Ⓜ] , toString [Ⓜ] , wait [Ⓜ] , wait [Ⓜ] , wait [Ⓜ]			

Constructor Details

Solver_Function_C
public Solver_Function_C(int Cap_Labor, int Cap_Grape, float Prc_Rose, float Prc_Noir) Initializes a new Solver_Function_C object that contains information on the capacity of Labor and Grape and the price of Ros [Ⓜ] and P-Noir.

Package src.main.java.function_B

Class Calculator

java.lang.Object
src.main.java.function_B.Calculator

```
public class Calculator
extends Object
```

Inheritance Tree

A Calculator class used to do all the calculation on optimization

Constructor Summary

Constructors	
Constructor	Description
Calculator()	

Method Summary

All Methods		
Instance Methods		Concrete Methods
Modifier and Type	Method	Description
int	Calculation(int Num_Rose, int Num_Noir, float prc_rose, float prc_noir)	Calculation based multiplication and addition to calculate the revenue
int[]	Optimization(int cap_labor, int cap_grape, float prc_rose, float prc_noir)	Calculate the optimize solution of opt_Rose,opt_Noir

Methods inherited from class java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

PPC1 Calculator

Annual Forecast by Gross Profit

Weekly Review by Revenue

Weekly Review by Revenue & Backorders

EXIT

Menu

A2.2.A Annual Forecast by Gross Profit

DATA INPUT	OPTIMIZED RESULT
Number of weeks (< 16): <input type="text"/>	Production Volume (Litre)
Capacity Of:	Rose: <input type="text"/>
Labor (minutes): <input type="text"/>	P-Noir: <input type="text"/>
Grape (Kgs): <input type="text"/>	Total: <input type="text"/>
Price of Product:	Total Gross Profit (A\$): <input type="text"/>
Rose (A\$/Litre): <input type="text"/>	Profit Margin/Revenue (%): <input type="text"/>
P-Noir (A\$/Litre): <input type="text"/>	
Total fixed costs of the year (A\$): <input type="text"/>	

System Messages / Warnings / Reminders

EXIT

RUN

branch A

A2.2.B Weekly Review By Revenue	
DATA INPUT	OPTIMAL RESULT
Number of weeks (2301-15): <input type="text"/>	Production Volume (Litre) Rose: <input type="text"/> P-Noir: <input type="text"/>
Capacity Of: Labor (minutes): <input type="text"/>	Total: <input type="text"/>
Grape (Kgs): <input type="text"/>	Revenue (A\$): <input type="text"/>
Price of Product:	Surplus: Labor (Minutes) <input type="text"/>
Rose (A\$/Litre): <input type="text"/>	Grape (Kgs) <input type="text"/>
P-Noir (A\$/Litre): <input type="text"/>	Total: <input type="text"/>
System Messages / Warnings / Reminders 	
<input type="button" value="Exit"/> <input type="button" value="Run"/>	

— □ ×

A2.2.C Weekly Review by Revenue & Backorders

DATA INPUT	OPTIMIZED RESULT
Week Of Year (2301-2315): <input style="width: 100px;" type="text"/>	
Capacity Of Labor (mins): <input style="width: 100px;" type="text"/>	Production Volume (L)
Capacity Of Grape (Kgs): <input style="width: 100px;" type="text"/>	Rose: <input style="width: 100px;" type="text"/>
Price Of Rose (A\$/L): <input style="width: 100px;" type="text"/>	P-Noir: <input style="width: 100px;" type="text"/>
Price Of P-Noir (A\$/L): <input style="width: 100px;" type="text"/>	<hr style="width: 100%;"/>
Backorder Volume Of Rose (L): <input style="width: 100px;" type="text"/>	Total: <input style="width: 100px;" type="text"/>
Backorde Volume Of P-Noir (L): <input style="width: 100px;" type="text"/>	Revenue (A\$): <input style="width: 100px;" type="text"/>
	Backorder Fulfillment: <input style="width: 100px;" type="text"/>

System Messages / Warnings / Reminders

EXIT
RUN